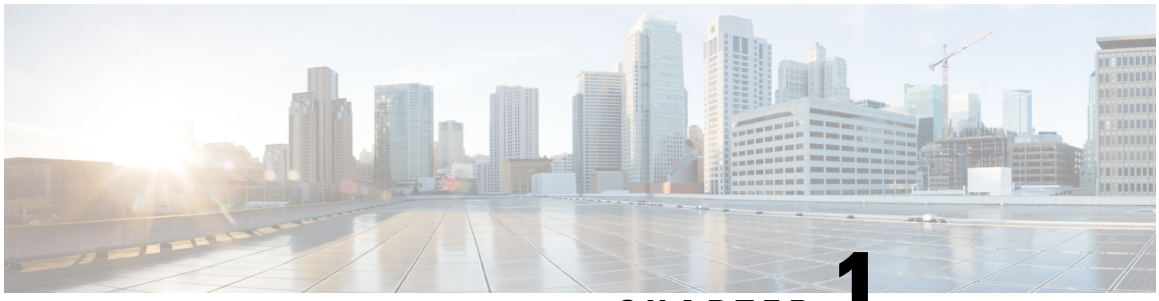
Cisco Enterprise Network Function Virtualization Infrastructure Software Configuration Guide, Release 3.12.x

Last Modified: 2022-03-15

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



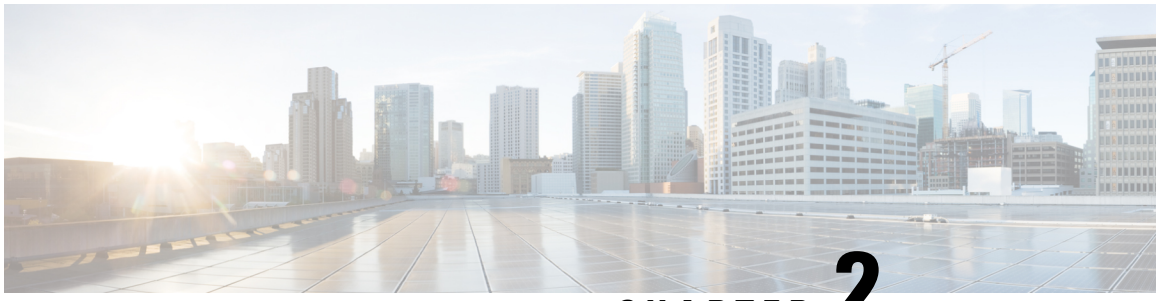
CHAPTER 1

New and Changed Information

The following table summarizes the new and changed features and tells you where they are documented.

Table 1: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 3.12.3

Feature	Description	Where Documented
Secure Tunnel enhacements for PKI and EAP	EAP and PKI authentication supported on secure overlay.	Secure Overlay, on page 117
APC UPS support	This feature provides support for monitoring battery status for an APC UPS connected to the ENCS box through a USB cable.	APC UPS Support and Monitoring, on page 58
SNMP support for CPU usage	Supported SNMP MIBs for CPU usage	SNMP MIB Support, on page 185
AAA auth-order	In this feature the supported aaa authentication order is local authentication followed by TACACS+.	Default Authentication Order, on page 24



CHAPTER 2

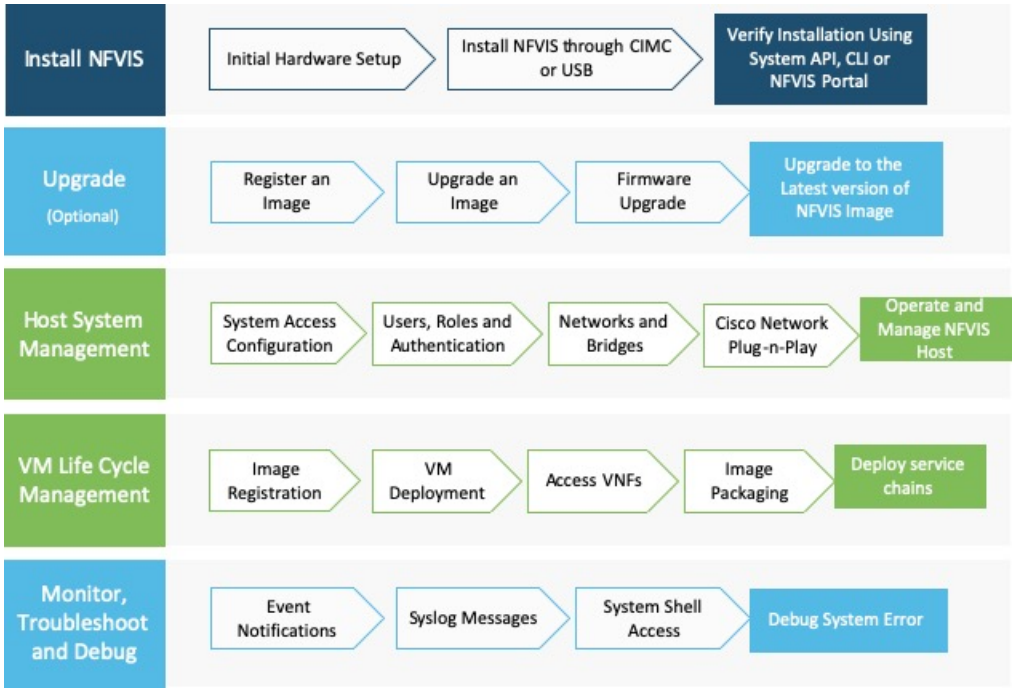
About NFVIS Documentation

Cisco Enterprise Network Function Virtualization Infrastructure Software (Cisco Enterprise NFVIS) is designed to help service providers and enterprises to design, deploy and manage network services. For more information about Cisco NFVIS, hardware platforms and VMs supported on it see, About Cisco Enterprise NFVIS.

This chapter describes how documentation about Cisco NFVIS is structured.

NFVIS Workflow

Typically, enterprises and service providers would go through the following stages in their NFVIS journey:

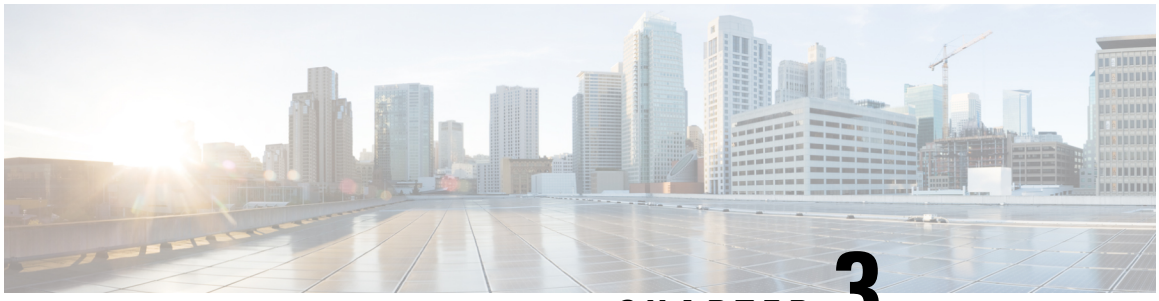


NFVIS documentation are structured according to the stages an enterprise or service provider would go through when they decide to setup NFVIS:

- The Getting Started Guide follows the same flow as enterprise or service providers would, at the time of deploying Virtual Network Functions (VNFs) for the first time.

- Set Up Cisco Enterprise NFVIS: Provides detailed information on getting started with ENCS 5400 Series platform devices.
- Install Cisco Enterprise NFVIS: Provides information on how to install Cisco NFVIS through Cisco IMC and USB for the supported hardware platforms.
- Upgrade Cisco NFVIS: Provides information on how to upgrade Cisco NFVIS to the latest version of the release.

- The Configuration Guide provides detailed information once you have completed the basic installation and set up. This guide covers managing the host system, registering and managing VNFs, security cosiderations for your network, troubleshooting issues with NFVIS and so on.
  - Host System Management: Provides detailed information about operations and management of NFVIS host.
  - VM Life Cycle Management, on page 63: Provides information on the entire process of registering, deploying, updating, monitoring VMs, and getting them service chained as per your requirements.
  - Troubleshoot and Debug Cisco NFVIS, on page 199: Information provided here helps troubleshoot and debug system errors.



CHAPTER 3

Host System Management

System Access Configuration

Dual WAN Support

Dual WAN support is introduced to provide multiple links to NFVIS connectivity. Starting from NFVIS 3.10.1 release, a second WAN bridge configured with DHCP by default is supported on ENCS 5000 series platform.

During NFVIS system initialization, NFVIS attempts to establish connectivity through DHCP on both WAN bridges. This allows connectivity to NFVIS during initial deployment even if the network is down on one of the WAN bridges. Once DHCP assigns an IP address through one WAN bridge, the other WAN bridge can be configured with static IP address for connectivity to NFVIS.

Restrictions for Dual WAN Support

- The DHCP toggle behavior is not supported during the upgrade flow. It is only triggered during fresh installation of NFVIS or after a factory default reset.
- Does not support active/standby or redundant WAN bridges. NFVIS does not detect connectivity failure from one WAN bridge and switchover to another WAN bridge. In case connectivity fails on the WAN bridge with DHCP configurations, connectivity through the other WAN bridge is established only if

static IP is applied to the second WAN bridge and static routing is configured for connectivity through that bridge.

- IPv6 is not supported for dual WAN toggle.
- If wan2-br is DHCP enabled WAN bridge, you must remove DHCP from wan2-br to apply default gateway from static IP configurations.

Dual WAN Bridge and DHCP Toggle

Note This feature is supported only on ENCS 5000 series devices.

In zero touch deployment, NFVIS requests for IPv4 assignments through DHCP for two WAN interfaces. During system initialization a second WAN bridge is configured with GE0-1 port attached. NFVIS toggles between the two default WAN bridges sending DHCP requests on any one of the WAN bridges at a time, for 30 second intervals. The toggling stops as soon as one WAN bridge is assigned an IP address through DHCP. The bridge with the assigned IP address is configured with DHCP. The other WAN bridge has no default IP configuration and can be manually configured with static IP if required.

If neither of the bridges is assigned an IP address through DHCP, the WAN DHCP toggle can be terminated by logging in to NFVIS using the default credentials. In this case, wan-br is configured with DHCP and wan2-br has no default IP configuration.

After zero touch deployment, the toggle feature is terminated. To add additional connectivity to the NFVIS host, static IP address can be configured on the other WAN bridge and system static routing can be applied. A default gateway is not supported as the system default gateway is set through DHCP. If DHCP configuration is not required, then both WAN bridges can be configured with static IP addresses, and a default gateway can then be applied under system settings.

Accessing NFVIS

For initial login, use **admin** as the default user name, and **Admin123#** as the default password. Immediately after the initial login, the system prompts you to change the default password. You must set a strong password as per the on-screen instructions to proceed with the application. All other operations are blocked until default password is changed. API will return 401 unauthorized error if the default password is not reset.

If wan-br or wan2-br have not obtained IP addresses through DHCP, the zero touch deployment is terminated. To manually apply the IP configurations answer 'y' and the system proceeds with DHCP assignment on wan-br until the configurations are changed. For DHCP assignment to continue to request IP address for PnP flow on both WAN interfaces answer 'n'.

You must adhere to the following rules to create a strong password:

- Must contain at least one upper case and one lower case letter.
- Must contain at least one number and one special character (# _ - * ?).
- Must contain seven characters or greater. Length should be between 7 and 128 characters.

You can change the default password in three ways:

- Using the Cisco Enterprise NFVIS portal.

- Using the CLI—When you first log into Cisco Enterprise NFVIS through SSH, the system will prompt you to change the password.
- Using PnP (for details, see the Cisco Network Plug-n-Play Support , on page 40).
- Using console - After the initial login using the default password, you are prompted to change the default password.

```
NFVIS Version: 3.12.3

Copyright (c) 2015-2020 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.

login: admin
NFVIS service is OK
Warning: Permanently added 'localhost' (RSA) to the list of known hosts.
admin@localhost's password:


Cisco Network Function Virtualization Infrastructure Software (NFVIS)

NFVIS Version: 3.12.3-RC8

Copyright (c) 2015-2020 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.


admin connected from ::1 using ssh on nfvis
admin logged with default credentials
Setting admin password will disable zero touch deployment behaviors.
Do you wish to proceed? [y or n]y
Please provide a password which satisfies the following criteria:
        1.At least one lowercase character
        2.At least one uppercase character
        3.At least one number
        4.At least one special character from # _ - * ?
        5.Length should be between 7 and 128 characters
Please reset the password :
Please reenter the password :


 Resetting admin password


New admin password is set

nfvis#
System message at 2020-01-08 03:10:10...
Commit performed by system via system using system.
nfvis#
```

Note To commit the target configuration to the active (running) configuration, use the **commit** command in any configuration mode. Changes made during a configuration session are inactive until the **commit** command is entered. By default, the commit operation is pseudo-atomic, meaning that all changes must succeed for the entire commit operation to succeed.

Connecting to the System

Using IPv4

The three interfaces that connect the user to the system are the WAN and WAN2 interfaces and the management interface. By default, the WAN interface has DHCP configuration and the management interface is configured with static IP address 192.168.1.1. If the system has a DHCP server connected to the WAN interface, the WAN interface is assigned an IP address from this server. You can use this IP address to connect to the system.

You can connect to the server locally (with an Ethernet cable) using the static management IP address; to connect to the box remotely using a static IP address, the default gateway needs to be configured.

You can connect to the system in the following three ways:

- Using the local portal—After the initial login, you are prompted to change the default password.
- Using the KVM console—After the initial login using the default password, you are prompted to change the default password.
- Using PnP—After the initial provisioning through PnP, the configuration file pushed by the PNP server must include the new password for the default user (admin).

Using IPv6

IPv6 can be configured in static, DHCP stateful and Stateless Autoconfiguration (SLAAC) mode. By default, DHCP IPv6 stateful is configured on the WAN interface. If DHCP stateful is not enabled on the network, the router advertisement (RA) flag decides which state the network stays in. If the RA shows Managed (M) flag, then the network stays in DHCP mode, even if there is no DHCP server in the network. If the RA shows Other (O) flag, then the network switches from DHCP server to SLAAC mode.

SLAAC provides IPv6 address and default gateway. Stateless DHCP is enabled in the SLAAC mode. If the server has DNS and domain configured, then SLAAC also provides those values via stateless DHCP.

Performing Static Configuration without DHCP

Note Starting from NFVIS 3.10.1 release, for ENCS 5400 and ENCS 5100, wan2-br obtains an IP address from DHCP. To configure default gateway, first use **no bridges bridge wan2-br dhcp** command.

If you want to disable DHCP and use static configuration, initial configuration is done by setting the WAN IP address and/or management IP address, and the default gateway. You can also configure a static IP on a created bridge.

To perform initial configuration on the system without using DHCP:

```
configure terminal
system settings mgmt ip address 192.168.1.2 255.255.255.0
```

```
bridges bridge wan-br ip address 209.165.201.22 255.255.255.0
system settings default-gw 209.165.201.1
commit
```

Note When an interface is configured with a static IP address, DHCP is automatically disabled on that interface.

Now you can either use the management IP or WAN IP to access the portal.

To configure static IPv6 on the WAN interface:

```
configure terminal
system settings mgmt ipv6 address 2001:DB8:1:1::72/64
bridges bridge wan-br ipv6 address 2001:DB8:1:1::75/64
system settings default-gw-ipv6 2001:DB8:1:1::76
commit
```

Note When an interface is configured with a static IPv6 address, DHCP IPv6 is automatically disabled on that interface. There are three options for IPv6 - static, DHCP and SLAAC, out of which only one can be enabled at a time.

To configure DHCP on the WAN interface:

```
configure terminal
no system settings default-gw
system settings wan dhcp
commit
exit
hostaction wan-dhcp-renew
```

Note Starting from NFVIS 3.10.1, you can configure DHCP IPv6 on any bridge. You can only have one DHCP IPv6 bridge or management interface active at a time, and cannot have DHCP IPv6 and default IPv6 gateway or SLAAC IPv6 configured at the same time.

To configure DHCP IPv6 on the WAN interface:

```
configure terminal
no system settings default-gw-ipv6
system settings wan dhcp-ipv6
commit
exit
hostaction wan-dhcp-renew
```

Verifying Initial Configuration

The **show system settings-native** command is used to verify initial configuration. Use **show bridge-settings** and **show bridge-settings** *bridge_name* commands to verify the configuration for any bridge on the system.

Extract from the output of the **show system settings-native** command when both WAN and management interfaces have a static configuration:

```
system settings-native mgmt ip-info interface lan-br
system settings-native mgmt ip-info ipv4_address 192.168.1.2
system settings-native mgmt ip-info netmask 255.255.255.0
!
!
!
system settings-native mgmt dhcp disabled
system settings-native wan ip-info interface wan-br
system settings-native wan ip-info ipv4_address 209.165.201.22
system settings-native wan ip-info netmask 255.255.255.0
!
!
!
system settings-native wan dhcp disabled
!
!
system settings-native gateway ipv4_address 209.165.201.1
system settings-native gateway interface wan-br
```

Extract from the output of the **show system settings-native** command when the management interface has a DHCP configuration and the WAN interface has a static configuration:

```
system settings-native mgmt ip-info interface MGMT
system settings-native mgmt ip-info ipv4_address 192.168.1.2
system settings-native mgmt ip-info netmask 255.255.255.0
!
!
!
system settings-native mgmt dhcp enabled
system settings-native wan ip-info interface wan-br
system settings-native wan ip-info ipv4_address 209.165.201.22
system settings-native wan ip-info netmask 255.255.255.0
!
!
!
system settings-native wan dhcp disabled
```

Extract from the output of the **show system settings-native** command when the WAN interface has a DHCP configuration and the management interface has a static configuration:

```
system settings-native mgmt ip-info interface lan-br
system settings-native mgmt ip-info ipv4_address 209.165.201.2
system settings-native mgmt ip-info netmask 255.255.255.0
!
!
!
system settings-native mgmt dhcp disabled
system settings-native wan ip-info interface wan-br
system settings-native wan ip-info ipv4_address 209.165.201.22
system settings-native wan ip-info netmask 255.255.255.0
!
!
!
system settings-native wan dhcp enabled
```

Configuring VLAN for NFVIS Management Traffic

A VLAN is a method of creating independent logical networks within a physical network. VLAN tagging is the practice of inserting a VLAN ID into a packet header in order to identify which VLAN the packet belongs to.

You can configure a VLAN tag on the WAN bridge (wan-br) interface to isolate Cisco Enterprise NFVIS management traffic from VM traffic. You can also configure VLAN on any bridge on the system (wan2-br for ENCS5400 or ENCS 5100, and user-br for all systems)

By default, Wan bridge and LAN bridge are in trunk mode and allows all VLANs. When you configure native VLAN, you must also configure all the allowed VLANs at the same time. The native VLAN becomes the only allowed VLAN if you do not configure all the VLANs. If you want a network that allows only one VLAN, then create another network on top of wan-net and lan-net and make it access network.

Note You cannot have the same VLAN configured for the NFVIS management and VM traffc.

For more details on the VLAN configuration, see the Understanding and Configuring VLANs module in the Catalyst 4500 Series Switch Cisco IOS Software Configuration Guide.

Configuring System Routes

In addition to the default routes in the system, you can configure additional system routes. This configuration is specifically useful when certain destinations are not reachable through the default routes.

While you can create a route just by providing the destination and prefix length, a valid route requires that you specify either a device or a gateway or both.

To configure additional system routes:

```
configure terminal
system routes route  209.165.201.1 dev lan-br
commit
```

Verifying the System Routes Configuration

To verify the system routes configuration, use the **show system routes** command as shown below:

```
nfvis# show system routes
DESTINATION  PREFIXLEN  STATUS
---------------------------------
209.165.201.1  12 -
209.165.201.2  12 -
209.165.201.3  24 -
```

System Routes APIs and Commands

System Routes APIs	System Routes Commands
• /api/config/system/routes • /api/config/system/routes/route/<host destination,netmask>	• system routes route • show system routes

Configuring the IP Receive ACL

To filter out unwanted traffic, you can configure ip-receive-acl to block or allow certain traffic based on the IP address and service ports.

To configure the source network for Access Control List (ACL) access:

```
configure terminal
system settings ip-receive-acl 198.0.2.0/24
action accept priority 10
commit
```

Verifying the Trusted IP Connection

Use the **show running-config system settings ip-receive-acl** command to display the configured source network for ACL access to the management interface

```
nfvis# show running-config system settings ip-receive-acl
system settings ip-receive-acl 198.51.100.11/24
service
[ ssh https scpd]
action accept
priority 100
```

Port 22222 and Management Interface ACL

Port 22222 is used for SCP server and is closed by default on an NFVIS system. You cannot SCP a file into NFVIS from an external server. If you need to SCP file from an external server, you must first open the port.

To open port 22222:

```
config terminal
system settings ip-receive-acl address/mask_len service scpd priority 2 action accept
commit
```

The Access Control List (ACL) is identify by address. If this ACL is removed, all ACLs sharing the same address are also removed. Ensure that you configure the ACLs that share the same address once again.

Note From 3.8.1 release, only a user belonging to administrator role can use the SCP command on this port to upload or download only from restricted folders like /data/intdatastore/. For more information, see Host System Operations, on page 50.

Caution SCP command cannot be used to copy files from one NFVIS device to another NFVIS device.

Use the **show running-config system settings ip-receive-acl** command to verify the interface configuration:

```
nfvis# show running-config system settings ip-receive-acl

system settings ip-receive-acl 10.156.0.0/16

 service  [ ssh https scpd ]
```

```
 action   accept
 priority 100
!
```

Configuring Secondary IP and Source Interface

Secondary IP

The Cisco Enterprise NFVIS supports multiple IP addresses per interface. A Secondary IP feature can be configured on the WAN interface, as an additional IP to reach the software. Set the external routes for Secondary IP to reach the NFVIS. Routers configured with secondary addresses can route between the different subnets attached to the same physical interface.

To access secondary IP through ISRv, the WAN physical port is removed from wan-br similar to single IP.

To configure Secondary IP:

```
Configure Secondary IP
nfvis(config)# system settings wan secondary ip address 1.1.2.3 255.255.255.0
```

Source Interface

This feature is used to set the source interface with an ip address. The ip address configured will be used for for packets generated by the NFVIS. The packets generated use the default route.

Prerequisites for configuring Source Interface

- IP must be one of the configured IP addresses in system settings.
- The source-interface IP address can be one of the following:
  - mgmt
  - WAN
  - WAN Secondary IP
  - WAN2 IP or IP configured on any bridge
- Source-interface configuration must be applied if the WAN IP is static.
- For DHCP, Source-interface IP is accepted but cannot be applied. The configuration takes effect once you switch from DHCP to static.

To configure Source Interface:

```
Configure source-interface ip
nfvis(config)# system settings source-interface
1.1.2.3
```

The Secondary IP and Source Interface related errors are logged in **show log nfvis_config.log** file.

Secondary IP and Source Interface APIs and Commands

APIs	Commands
• /api/config/system/settings/wan/secondary	• system settings wan secondary
• /api/config/system/settings/source-interface	• system settings source-interface

CIMC Access Control

On ENCS 5400, NFVIS administrators have authoritative control of the device. This includes capability to change the IP address used to reach the CIMC and modifying the CIMC and BIOS passwords

CIMC Access using NFVIS

Note CIMC access using NFVIS is supported only on ENCS 5400.

When CIMC access is enabled on NFVIS, ISRv can gain access to the host CIMC and internal switch management console. You must have authorization from Cisco Interactive Debug (CID) to access both consoles.

To access CIMC using NFVIS WAN or management interface IP address, use the **system settings cimc-access enable** command. Once you configure CIMC access on NFVIS, the stand alone CIMC access using CIMC IP address is disabled and you will be able to access CIMC using NFVIS management interface IP address. The configurations remain on the device even after the device reboot.

When the CIMC access is configured, it enables a few ports to access services like SSH, SNMP, HTTP and HTTPs into the CIMC.

The following port numbers are being used for forwarding services to CIMC:

- 20226 for SNMP
- 20227 for SSH
- 20228 for HTTP
- 20229 for HTTPS

If you are unable to access CIMC using NFVIS, check the show log nfvis_config.log file.

Use **system settings cimc-access disable** to disable this feature.

BIOS-CIMC Update

Starting from 3.8.1 release, for ENCS 5400 router, if existing BIOS/CIMC version is lower than the bundled image in NFVIS ISO or upgrade package, it is updated automatically during the NFVIS upgrade or installation. Also the CPU microcode is upgraded. The upgrade time takes longer than the previous releases and the upgrade will be done automatically, and you cannot stop the process once it is initiated.

For ENCS 5100 router, BIOS will be upgraded automatically to a new version but you need to boot up the server manually after the upgrade.

BIOS and CIMC Password

To change the BIOS and CIMC password for ENCS 5400 use **hostaction change-bios-password newpassword** or **hostaction change-cimc-password newpassword** commands. The change in the password will take effect immediately after the commands are executed. For both CIMC and BIOS passwords any alphanumeric character along with some special characters (_ @ #) are allowed.

For CIMC, the password must contain a minimum of eight characters..

For BIOS, the password must contain a minimum of seven characters and the first letter cannot be #.

BIOS and CIMC Password APIs and Commands

BIOS and CIMC Password APIs	BIOS and CIMC Password Commands
• /api/operations/hostaction/change-cimc-password /api/operations/hostaction/change-bios-password	• hostaction change-cimc-password • hostaction change-bios-password

Overview to ENCS 5400 for UEFI Secure Boot

You can use Unified Extensible Firmware Interface (UEFI) secure boot to ensure that all the EFI drivers, EFI applications, option ROM or operating systems prior to loading and execution are signed and verified for authenticity and integrity, before you load and execute the operating system. You can enable this option using either web UI or CLI. When you enable UEFI secure boot mode, the boot mode is set to UEFI mode and you cannot modify the configured boot mode until the UEFI boot mode is disabled.

Note If you enable UEFI secure boot on a nonsupported OS, on the next reboot, you cannot boot from that particular OS. If you try to boot from the previous OS, an error is reported and recorded the under system software event in the web UI. You must disable the UEFI secure boot option using Cisco IMC to boot from your previous OS.

Enabling UEFI Secure Boot Mode

To enable UEFI secure boot mode:

```
Server# scope bios
Server /bios # set secure-boot enable
Setting Value : enable
Commit Pending.
Server /bios *# commit
```

Reboot the server to have your configuration boot mode settings take place.

Disabling UEFI Secure Boot Mode

To disable UEFI secure boot mode:

```
Server# scope bios
Server /bios # set secure-boot disable
Setting Value : enable
```

```
Commit Pending.
Server /bios *# commit
```

Reboot the server to have your configuration boot mode settings take place.

To install NFVIS in UEFI mode, map the iso image through vmedia or kvm first, then enable secure boot and change the BIOS set-up parameters.

```
encs# scope bios
encs /bios # scope advanced
encs /bios/advanced # set BootOpRom UEFI
encs /bios/advanced # set BootOrderRules Loose
encs /bios/advanced *# commit
```

Reboot the device to start installation.

Note All VNFs and configurations are lost at reboot. Secure boot in UEFI mode works differently from the legacy mode. Therefore, there is no compatibility in between legacy mode and UEFI mode. The previous environment is not kept.

Enabling or Disabling the Portal Access

The Cisco Enterprise NFVIS portal access is enabled by default. You can disable the access if required.

To disable the portal access:

```
configure terminal
system portal access disabled
commit
```

Note You can enable the portal access using the **enabled** keyword with the **system portal access** configuration.

Verifying the Portal Access

Use the **show system portal status** command to verify the portal access status as shown below:

```
nfvis# show system portal status
system portal status "access disabled"
```

Portal Access APIs and Commands

Portal Access APIs	Portal Access Commands
• /api/config/system/portal • /api/operational/system/portal/status	• system portal access • show system portal status

Users, Roles and Authentication

Local User Account Management

Role based access enables the administrator to manage different levels of access to the system's compute, storage, database, and application services. It uses the access control concepts such as users, groups, and rules, which you can apply to individual API calls. You can also keep a log of all user activities.

Table 2: Supported User Roles and Privileges

User Role	Privilege
Administrators	Owns everything, can perform all tasks including changing of user roles, but cannot delete basic infrastructure. Admin's role cannot be changed; it is always "administrators".
Operators	Start and stop a VM, and view all information
Auditors	Read-only permission

Rules for User Passwords

The user passwords must meet the following requirements:

- Must have at least seven characters length or the minimum required length configured by the admin user.
- Must not have more than 128 characters.
- Must contain a digit.
- Must contain one of the following special characters: hash (#), underscore (_), hyphen (-), asterisk (*), and question mark (?).
- Must contain an uppercase character and a lowercase character.
- Must not be same as last five passwords.

Creating Users and Assigning Roles

The administrator can create users and define user roles as required. You can assign a user to a particular user group. For example, the user "test1" can be added to the user group "administrators".

Note All user groups are created by the system. You cannot create or modify a user group.

Starting from NFVIS 3.9.1, create-user, delete-user, change-role and change-password operations are configurable from exec mode.

To create a user:

```
rbac authentication users create-user name test1 password Test1_pass role administrators
```

To delete a user:

```
rbac authentication users delete-user name test1
```

Note To change the password, use the **rbac authentication users user test1 change-password new-password newPassword old-password oldPassword** command. To change the user role to administrators, operators or auditors, use the **rbac authentication users user test1 change-role new-role newRole old-role oldRole** command.

User Management APIs and Commands

User Management APIs	User Management Commands
• /api/config/rbac/authentication/users • /api/operations/rbac/authentication/users /user/<user-name>/change-password • /api/operations/rbac/authentication/users/user /<user-name>/change-role • /api/operations/rbac/authentication/users/create-user • /api/operations/rbac/authentication/users/delete-user	• rbac authentication users • rbac authentication users user <user-name> change-password old-password <old_pwd> new-password <new_pwd> • rbac authentication users user <user-name> change-role old-role <old_role> new-role <new_role> • rbac authentication users create-user name <user-name> password <password> role <role> • rbac authentication users delete-user name <user-name>

Configuring Minimum Length for Passwords

The admin user can configure the minimum length required for passwords of all users. The minimum length must be between 7 to 128 characters. By default, the minimum length required for passwords is set to 7 characters.

```
configure terminal
rbac authentication min-pwd-length 10
commit
```

Minimum Password Length APIs and Commands

APIs	Commands
/api/config/rbac/authentication/min-pwd-length	rbac authentication min-pwd-length

Configuring Password Lifetime

The admin user can configure minimum and maximum lifetime values for passwords of all users and enforce a rule to check these values. The default minimum lifetime value is set to 1 day and the default maximum lifetime value is set to 60 days.

When a minimum lifetime value is configured, the user cannot change the password until the specified number of days have passed. Similarly, when a maximum lifetime value is configured, a user must change the password before the specified number of days pass. If a user does not change the password and the specified number of days have passed, a notification is sent to the user.

Note The minimum and maximum lifetime values and the rule to check for these values are not applied to the admin user.

```
configure terminal
rbac authentication password-lifetime enforce true min-days 2 max-days 30
commit
```

Password Lifetime APIs and Commands

APIs	Commands
/api/config/rbac/authentication/password-lifetime/	rbac authentication password-lifetime

Deactivating Inactive User Accounts

The admin user can configure the number of days after which an unused user account is marked as inactive and enforce a rule to check the configured inactivity period. When marked as inactive, the user cannot login to the system. To allow the user to login to the system, the admin user can activate the user account by using the **rbac authentication users user** *username* **activate** command.

Note The inactivity period and the rule to check the inactivity period are not applied to the admin user.

```
configure terminal
rbac authentication account-inactivity enforce true inactivity-days 2
commit
```

Deactivate Inactive User Accounts APIs and Commands

APIs	Commands
/api/config/rbac/authentication/account-inactivity/	rbac authentication account-inactivity

Activating an Inactive User Account

The admin user can activate the account of an inactive user.

```
configure terminal
```

```
rbac authentication users user guest_user activate
commit
```

Activate Inactive User Account APIs and Commands

APIs	Commands
/api/operations/rbac/authentication/users/user/username/activate	rbac authentication users user activate

NFVIS Password Recovery

1. Load the NFVIS ISO image, using the CIMC KVM console.
2. Select Troubleshooting from the Boot Selection menu.
3. Select Rescue a NFVIS Password.
4. Select Continue.
5. Press Return to get a shell.
6. Run the **chroot /mnt/sysimage** command.
7. Run the **./nfvis_password_reset** command to reset the password to admin.
8. Confirm the change in password and enter Exit twice.

   Disconnect the NFVIS ISO image in the CIMC KVM console and reboot NFVIS.
9. Login to NFVIS with the default credentials admin/Admin123#.

   After login to NFVIS, enter a new password at prompt.
10. Connect to NFVIS with the new password.

Note You can update and recover NFVIS 3.8.1 and older passwords using NFVIS 3.9.1.

RADIUS Support

-

About RADIUS

Remote Authentication Dial-In User Service (RADIUS) is a distributed client-server system that secures networks against unauthorized access. In the Cisco implementation, RADIUS clients run on Cisco routers and send authentication requests to a central RADIUS server that contains all user authentication and network service access information.

RADIUS is a fully open protocol, distributed in source code format, that can be modified to work with any security system currently available on the market.

Cisco supports RADIUS under its AAA security paradigm. RADIUS has been implemented in a variety of network environments that require high levels of security while maintaining network access for remote users.

RADIUS Operation

When a user attempts to log in and authenticate to an access server using RADIUS, the following steps occur:

1. The user is prompted to enter the username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of the following responses from the RADIUS server:
   a. ACCEPT—The user is authenticated.
   b. CHALLENGE—A challenge is issued by the RADIUS server. The challenge collects additional data from the user.
   c. CHANGE PASSWORD—A request is issued by the RADIUS server, asking the user to select a new password.
   d. REJECT—The user is not authenticated and is prompted to reenter the username and password, or access is denied.

The ACCEPT or REJECT response is bundled with additional data that is used for EXEC or network authorization. You must first complete RADIUS authentication before using RADIUS authorization. The additional data included with the ACCEPT or REJECT packets consists of the following:

- Services that the user can access, including connections such as Telnet, rlogin, or local-area transport (LAT), and services such as PPP, Serial Line Protocol (SLIP), or EXEC services.
- Connection parameters, including the host or client IP address, access list, and user timeouts.

Configuring RADIUS

To configure RADIUS support:

```
configure terminal
radius-server host 103.1.4.3
shared-secret cisco123
admin-priv 15
oper-priv  11
commit
```

Starting from NFVIS 3.9.2 release, RADIUS secret encryption is supported. You can only configure either secret key or encrypted secret key at a given time. Use encrypted secret if special characters are used in secret. To configure encrypted RADIUS secret:

```
configure terminal
radius-server host 103.1.4.3
encrypted-shared-secret cisco123
admin-priv 15
oper-priv  11
commit
```

Verifying the RADIUS configuration

Use the **show running-config radius-server** command to verify the interface configuration for a RADIUS session:

```
nfvis# show running-config radius-server

radius-server host 103.1.4.3
 key           0
 shared-secret cisco123
 admin-priv    15
 oper-priv     11
```

RADIUS Support APIs and Commands

APIs	Commands
• /api/config/security_servers/radius-server	• radius-server host • key • admin-priv • oper-priv • encrypted-shared-secret or shared-secret

TACACS+ Support

•

About TACACS+

TACACS+ is a security application that provides centralized validation of users attempting to gain access to a router or network access server. TACACS+ services are maintained in a database on a TACACS+ daemon running, typically, on a UNIX or Windows NT workstation. You must configure a TACACS+ server before the configured TACACS+ features on your network access server are available.

On the TACACS+ server, ensure you configure Cisco attribute-value (AV) pair privilege level (priv-lvl) for Cisco Enterprise NFVIS service for the minimum privilege level of administrators and operators.

Note In NFVIS 3.11.1 or earlier release, users with no privilege level or users with a privilege level that is less than the operator's privilege level are considered as auditors with read-only permission.

After NFVIS 3.12.1 release, users with privilege level zero won't be able to login to NFVIS anymore.

TACACS Operation

When a user attempts a simple ASCII login by authenticating to NFVIS using TACACS+, this process occurs:

1. When the user tries to log in, NFVIS sends user credential to TACACS+ server.
2. NFVIS will eventually receive one of the following responses from the TACACS+ server:
   a. ACCEPT—The user is authenticated and service can begin. If NFVIS is configured to require authorization, authorization begins at this time.

b. REJECT—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ server.

c. ERROR—An error occurred at some time during authentication with the server or in the network connection between the server and NFVIS. If an ERROR response is received, NFVIS typically tries to use an alternative method for authenticating the user.

d. CONTINUE—The user is prompted for additional authentication information.

After authentication, NFVIS will send authorization request to TACACS+ server.

3. Based on authorization result, NFVIS will assign user's role.

Configuring a TACACS+ Server

To configure TACACS+:

```
configure terminal
tacacs-server host 209.165.201.20 shared-secret
test1

key 0
admin-priv
14

oper-priv
9

commit
```

In this configuration, privilege level 14 is assigned to the administrator role, and privilege level 9 is assigned to the operator role. This means a user with privilge level 14 or higher will have all admin privileges when the user logs into the system, and a user with privilege level 9 or higher will have all privileges of an operator at the time of login.

Starting from NFVIS 3.9.2 release, TACACS+ secret encryption is supported. You can only configure either secret key or encrypted secret key at a given time. Encrypted secret key can contain special characters but secret key cannot. For NFVIS 3.12.1 release, the following pattern is supported for encryped-shared-key: [-_a-zA-Z0-9./\\<>%!*$€#{}()+].

To configure encrypted TACACS+ key:

```
configure terminal
tacacs-server host 209.165.201.20 encrypted-shared-secret test1
key 0
admin-priv
14

oper-priv
9

commit
```

Verifying the TACAC+ configuration

Use the **show running-config tacacs-server** command to verify the configuration if encrypted TACACS+ key is configured:

```
nfvis# show running-config tacacs-server

tacacs-server host 209.165.201.20
 encrypted-shared-secret $8$mRTnL9TKZCFi1BUP7Mwbm3JVIo4Z7QvJ
 admin-priv              15
 oper-priv               11
!
```

TACACS+ APIs and Commands

TACACS+ APIs	TACACS+ Commands
• /api/config/security_servers/tacacs-server • /api/config/security_servers/tacacs-server?deep • /api/config/security_servers/tacacs-server /host/<ip-address/domain-name>	• tacacs-server host • key • admin-priv • oper-priv • encrypted-shared-secret or shared-secret

Default Authentication Order

NFVIS supports both TACACS+ and RADIUS but only one authentication method can be enabled at a time. After you have identified the TACACS+ and RADIUS server and defined an associated TACACS+ and RADIUS authentication key, you must define method lists for TACACS+ and RADIUS authentication. Because TACACS+ and RADIUS authentication is operated through AAA, you need to issue the aaa authentication command, specifying TACACS+ or RADIUS as the authentication method.

```
nfvis(config)# aaa authentication ?
Possible completions:
  radius   Use RADIUS for AAA
  tacacs   Use TACACS+ for AAA
  users    List of local users
```

Note

- Only when TACACS+ or RADIUS is enabled, it can be used for authentication.
- When TACACS+ or RADIUS is not accessible, local authentication is used. It is recommended to use **aaa authentication TACACS local** command to authenticate using local database. Local authentication is disabled if the connection between TACACS+ or RADIUS and NFVIS is restored.
- If same username exists on both local and TACACS+ or RADIUS, then TACACS+ or RADIUS user is chosen for authentication.
- It is recommended to configure Syslog, on page 179 so that it is easier to debug if TACACS+ or RADIUS does not work as expected.

All login attempts will be logged in syslogs in the local *nfvis_syslog.log*, *nfvis-ext-auth.log* files and in remote syslog servers.

Enchancements for NFVIS 3.12.3, User Specific Authentication Order

Starting from NFVIS 3.12.3 release, the supported aaa authentication order is local authentication followed by TACACS+.

- If the user is in local database execute authentication and permit or deny access.
- If the user is not in local database, use TACACS+ for authentication
- If the same user is present in both local and TACACS+, then the user can login with local password or TACACS+ password. It is not recommended to configure same user in both local and TACACS+.

In NFVIS 3.12.3 release the only supported combination for authentication order is **aaa auth-order local tacacs**. Any other combinations are not supported. **aaa auth-order** configuration is mutually exclusive to **aaa authentication** and if one is configured, the other is automatically replaced.

```
nfvis(config)# aaa ?
Possible completions:
auth-order Configure authentication order; Mutually exclusive to authentication method
configuration
authentication Configure external authentication method; Mutually exclusive to auth-order
configuration
ios Specific IOS settings

nfvis(config)# aaa auth-order ?
Description: Configure authentication order; Mutually exclusive to authentication method
configuration
Possible completions:local radius tacacs
nfvis(config)# aaa auth-order local ?
Possible completions:
radius tacacs <cr>
nfvis(config)# aaa auth-order local tacacs ?
Possible completions:radius <cr>
nfvis(config)# aaa auth-order local tacacs
nfvis(config)#
```

Networking

-

Bridges

The IP configuration on bridges and **show bridge-settings** command were added in NFVIS 3.10.1 release. NFVIS is installed with LAN and WAN bridges by default. A service bridge can also be created. A bridge can be used for NFVIS connectivity. Each bridge can be configured with IPv4 or IPv6 configurations such as Static IP, DHCP, SLAAC, or VLAN. Each bridge can have a port or port channel associated with it.

On all NFVIS systems, lan-br and wan-br are generated by default and populated with the appropriate ports for that system. On ENCS 5000 series platforms wan2-br is also generated by default for the dual WAN initialization. For more information, see Dual WAN Support, on page 5. Except on ENCS 5000 Series platforms, the default LAN bridge is configured with a static IP address 192.168.1.1 and the WAN bridges uses DHCP for initial NFVIS connectivity.

On ENCS 5400 series platforms configuration changes are not allowed on the lan-br bridge. The LAN bridge cannot be modified in any way.

Using IPv4

If the system has a DHCP server connected to a bridge with DHCP configured, the bridge receives the IP address from the server. You can use this IP address to connect to the system.

You can also connect to the server locally with an ethernet cable using a static IP address. To connect to the box remotely using a static IP address, you must configure the default gateway or setup an appropriate static route.

Both DHCP and a default gateway cannot be configured on NFVIS simultaneously. NFVIS only supports one system level default gateway and if DHCP is configured, the default gateway is assigned to the system through the DHCP server. Also, only one bridge can be configured with DHCP at any time.

Using IPv6

IPv6 can be configured in static, DHCP stateful and Stateless Auto configuration (SLAAC) modes. By default,DHCP IPv6 stateful is configured on the WAN interface. If DHCP stateful is not enabled on the network, the router advertisement (RA) flag decides which state the network stays in. If the RA shows Managed (M) flag, then the network stays in DHCP mode, even if there is no DHCP server in the network. If the RA shows Other (O) flag, then the network switches from DHCP server to SLAAC mode.

SLAAC provides IPv6 address and a default gateway. Stateless DHCP is enabled in the SLAAC mode. If the server has DNS and domain configured, then SLAAC also provides those values through stateless DHCP.

Similar to IPv4, IPv6 DHCP and IPv6 default gateway cannot be configured on the system simultaneously, nor can stateful and stateless IPv6 DHCP. Also, only one bridge can be configured with either stateful or stateless IPv6 DHCP at any time.

Creating Bridges

To configure a new bridge:

```
configure terminal
bridges bridge my-br
commit
```

To verify the bridge generation, use the **show bridge-settings** command:

```
nfvis# show bridge-settings my-br ip-info interface
ip-info interface my-br
```

Configuring Bridge Port

A bridge can be tied to a physical interface by applying the port configuration. A bridge can have as many ports as are available, however a port must be unique to at most one bridge. If a port channel is applied to a bridge, it must be the only port configuration on that bridge.

To configure a port on a bridge:

```
configure terminal
bridges bridge my-br port eth3
commit
```

To configure a port channel on a bridge:

```
configure terminal
```

```
bridges bridge my-br port pc1
commit
```

To verify the port settings applied to a bridge, use the **support ovs vsctl** command:

```
nfvis# support ovs vsctl list-ports my-br
eth3
```

The same command can be used to verify the port channel settings applied to a bridge:

```
nfvis# support ovs vsctl list-ports my-br
bond-pc1
```

Configuring Bridge IP Connectivity

Configuring DHCP on Bridge

DHCP configuration can be applied to any bridge if no other bridge on the system has DHCP configured, and default gateway is not applied under system settings. Starting from NFVIS 3.12.1 release, DHCP configuration on a bridge automatically triggers a DHCP renew request from the bridge. For an additional DHCP renew trigger, use the **hostaction bridge-dhcp-renew** command.

To configure DHCP on a bridge:

```
configure terminal
bridges bridge my-br dhcp
commit
```

To verify the DHCP settings applied to a bridge, use the **show bridge-settings <br_name> dhcp** command.

```
nfvis# show bridge-settings my-br dhcp

dhcp enabled
dhcp offer                          true
dhcp interface                      my-br
dhcp fixed_address                  10.10.10.14
dhcp subnet_mask                     255.255.255.128
dhcp gateway                         10.10.10.1
dhcp lease_time                      7200
dhcp message_type                      5
dhcp name_servers                     NA
dhcp server_identifier               10.10.10.1
dhcp renewal_time                    3600
dhcp rebinding_time                    6300
dhcp vendor_encapsulated_options     NA
dhcp domain_name                       NA
dhcp renew                       2019-12-11T13:28:29-00:00
dhcp rebind                     2019-12-11T14:17:12-00:00
dhcp expire                     2019-12-11T14:32:12-00:00
```

Configuring Static IP on Bridge

An IPv4 address and subnet can be configured on any bridge which does not have DHCP configured. To enable routing outside of the subnet, apply the default gateway under system settings or confirgure system routes.

To configure an IPv4 address on a bridge:

```
configure terminal
bridges bridge my-br ip address 172.25.220.124 255.255.255.0
commit
```

To verify the IPv4 settings applied to a bridge, use the **show bridge-settings <br_name> ip_info** command.

```
nfvis# show bridge-settings my-br ip_info
ip-info interface                        my-br
ip-info ipv4_address                     172.25.220.124
ip-info netmask                          255.255.255.0
ip-info link-local ipv6 address          fe80::4e00:82ff:fead:e802
ip-info link-local ipv6 prefixlen        64
ip-info global ipv6                     address::
ip-info global ipv6 prefix               len0
ip-info mac_address                      4c:00:82:ad:e8:02
ip-info mtu                             9216
ip-info txqueuelen                      1000
```

Configuring IPv6 DHCP on Bridge

IPv6 DHCP configuration can be applied to any bridge if no other bridge on the system has IPv6 DHCP or IPv6 SLAAC configured, and IPv6 default gateway is not applied under system settings. Starting from NFVIS 3.12.1 release, an IPv6 DHCP configuration on a bridge automatically triggers an IPv6 DHCP renew request from the bridge. For an additional IPv6 DHCP renew trigger use the **hostaction bridge-dhcp-renew** command.

To configure IPv6 DHCP on a bridge:

```
configure terminal
bridges bridge my-br dhcp-ipv6
commit
```

To verify the IPv6 DHCP settings applied to a bridge, use the **show bridge-settings <br_name> dhcp-ipv6** command.

```
nfvis# show bridge-settings my-br dhcp-ipv6
dhcp-ipv6 offer true
dhcp-ipv6 interface my-br
dhcp-ipv6 ia-naec:d2:7d:b4
dhcp-ipv6 starts 1554792146
dhcp-ipv6 renew 43200
dhcp-ipv6 rebind 69120
dhcp-ipv6 iaaddr 2001:420:30d:201:ffff:ffff:fffa:8e48
dhcp-ipv6 preferred-life 86400
dhcp-ipv6 max-life 172800
dhcp-ipv6 client-id 0:1:0:1:24:3e:fb:50:0:62:ec:d2:7d:b4
dhcp-ipv6 server-id 0:3:0:1:0:25:45:1b:c2:2a
dhcp-ipv6 name_servers NA
dhcp-ipv6 domain_name NA
dhcp-ipv6 option [ ]
```

Configuring IPv6 SLAAC on Bridge

IPv6 SLAAC configuration can be applied to any bridge if no other bridge on the system has IPv6 SLAAC or IPv6 DHCP configured, and IPv6 default gateway is not applied under system settings.

To configure IPv6 SLAAC on a bridge:

```
configure terminal
```

```
bridges bridge my-br slaac-ipv6
commit
```

To verify the IPv6 SLAAC settings applied to a bridge, use the **show bridge-settings <br_name> slaac-ipv6** command.

```
nfvis# show bridge-settings my-br slaac-ipv6
slaac-ipv6 enabled
```

Configuring Static IPv6 Address on Bridge

An IPv6 address can be configured on any bridge which does not have IPv6 DHCP or SLAAC configured. To enable routing outside of the subnet, apply the default gateway under system settings or confirgure system routes.

To configure an IPv6 address on a bridge:

```
configure terminal
bridges bridge my-br ipv6 address 2001:db8:85a3::8a2e:370:7334/64
commit
```

To verify the IPv6 settings applied to a bridge, use the **show bridge-settings <br_name> ip_info** command.

```
nfvis# show bridge-settings my-br ip_info
ip-info interface my-br
ip-info ipv4_address 172.25.220.124
ip-info netmask 255.255.255.0
ip-info link-local ipv6 address fe80::4e00:82ff:fead:e802
ip-info link-local ipv6 prefixlen 64
ip-info global ipv6 address 2001:db8:85a3::8a2e:370:7334
ip-info global ipv6 prefixlen 64
ip-info mac_address 4c:00:82:ad:e8:02
ip-info mtu 9216
ip-info txqueuelen 1000
```

Configuring VLAN on Bridge

A VLAN is a method of creating independent logical networks within a physical network. VLAN tagging is the practice of inserting a VLAN ID into a packet header in order to identify which VLAN the packet belongs to.

You can configure a VLAN tag on the WAN bridge (wan-br) interface to isolate Cisco Enterprise NFVIS management traffic from VM traffic. You can also configure VLAN on any bridge on the system (wan2-br for ENCS5400 or ENCS 5100, and user-br for all systems)

By default, Wan bridge and LAN bridge are in trunk mode and allows all VLANs. When you configure native VLAN, you must also configure all the allowed VLANs at the same time. The native VLAN becomes the only allowed VLAN if you do not configure all the VLANs. If you want a network that allows only one VLAN, then create another network on top of wan-net and lan-net and make it access network.

Note You cannot have the same VLAN configured for the NFVIS management and VM traffc.

For more details on the VLAN configuration, see the Understanding and Configuring VLANs module in the Catalyst 4500 Series Switch Cisco IOS Software Configuration Guide.

To configure a VLAN:

```
configure terminal
bridges bridge wan-br vlan 120
commit
```

To verify the VLAN settings applied to a bridge, use the **show bridge-settings my-br vlan** command.

```
nfvis# show bridge-settings my-br vlan
vlan tag 10
```

Configuring MAC Aging Time on Bridge

MAC aging time specifies the time at which a MAC address entry ages out of the MAC address table. The max-aging-time specifies the maximum number of seconds to retain a MAC learning entry for which no packets have been seen. The default value is 300 seconds.

To configure MAC aging time on a bridge:

```
configure terminal
bridges bridge my-br mac-aging-time 600
commit
```

To verify the MAC aging time settings applied to a bridge, use the **show bridge-settings <br_name> mac-aging-time** command.

```
nfvis# show bridge-settings my-br mac-aging-time
mac-aging-time 600
```

Bridge APIs and Commands

Bridge APIs	Bridge Commands
/api/operational/bridge-settings /api/config/bridges/bridge/	bridges bridge <br_name> bridges bridge <br_name> port bridges bridge <br_name> ip address bridges bridge <br_name> dhcp bridges bridge <br_name> ipv6 address bridges bridge <br_name> dhcp-ipv6 bridges bridge <br_name> slaac-ipv6 bridges bridge <br_name> vlan bridges bridge <br_name> mac-aging-time show bridge-settings <br_name> support ovs vsctl list-ports <br_name>

Port Channels

Information About Port Channels

Port channels combine individual links into a group to create a single logical link that provides the aggregate bandwidth of up to eight physical links. Creating port channels helps to increase bandwidth and redundancy and to load balance traffic between the member ports. If a member port within a port channel fails, the traffic from the failed port switches to the remaining member ports.

Port channels must have atleast two ports and can be configured using static mode or Link Access Control Protocol (LACP). Configuration changes that are applied to the port channel are applied to each member port of the port channel. A port channel can also be added to a bridge. When a port channel has two or more than two members and the port channel is added to a bridge, a bond is created.

A port can be a member of only one port channel and all the ports in a port channel must be compatible. Each port must use the same speed and operate in full-duplex mode.

Port Channels Bond Mode

A port channel can be configured for the following bond modes:

- **`active-backup`**: In this mode, one of the ports in the aggregated link is active and all others ports are in the standby mode.
- **`balance-slb`**: In this mode, load balancing of traffic is done based on the source MAC address and VLAN.
- **`balance-tcp`**: In this mode, 5-tuple (source and destination IP, source and destination port, protocol) is used to balance traffic across the ports in an aggregated link.

Port Channels LACP Mode

A port channel can be configured for the following LACP modes:

- **`off`**: Indicates that no mode is applicable.
- **`active`**: Indicates that the port initiates transmission of LACP packets.
- **`passive`**: Indicates that the port only responds to the LACP packets that it receives but does not initiate the LACP negotiation.

Configuring a Port Channel

Creating a Port Channel

To create a port channel:

```
configure terminal
pnic egroup type port_channel lacp_type active bond_mode balance-tcp trunks 10,20
commit
```

Adding a Port to a Port Channel

You can add a port to a new port channel or a port channel that already contains ports. To add a port to a port channel:

Adding GE0-0 and GE0-1 to egroup:

```
configure terminal
pnic GE0-0 member_of egroup
commit

configure terminal
pnic GE0-1 member_of egroup
commit
```

Adding a Port Channel to a Bridge

You can add a port channel to a new bridge or an existing bridge. When a port channel is added to a bridge, a bond is added for the port channel.

To add a port channel to a bridge:

```
configure terminal
bridges bridge test-br port egroup
commit
```

Deleting a Port Channel

Before deleting a port channel, you must remove all members assigned to the port channel. If the port channel is configured on the bridge, you must remove the port channel from the bridge.

1. Remove ports from port channel. If GE0-0 and GE0-1 are part of port channel pc, remove them from pc first.

   ```
   configure terminal
   no pnic pc GE0-0 member_of egroup
   commit

   configure terminal
   no pnic GE0-1 member_of egroup
   commit
   ```

2. Remove port channel from the bridge.

   ```
   configure terminal
   no bridges bridge test-br port egroup
   commit
   ```

3. Delete port channel.

   ```
   configure terminal
   no pnic egroup
   commit
   ```

Verifying Port Channel Configurations

To verify port channel configurations, use the **show port-channel** command.

```
nfvis# show port-channel

----bond-egroup----
bond_mode: balance-tcp
bond may use recirculation: yes, Recirc-ID : 1
bond-hash-basis: 0
updelay: 0 ms
downdelay: 0 ms
next rebalance: 6921 ms
lacp_status: negotiated >>>this should be negotiated to indicate port channel is active
lacp_fallback_ab: false
active slave mac: 38:90:a5:1b:fe:0d(GE0-1)>>should indicate active slave mac address

slave GE0-0: enabled
may_enable: true

slave GE0-1: enabled
active slave >>>active slaveport should show active
may_enable: true
```

Port Channel APIs and Commands

APIs	Commands
/api/config/pnics /api/config/pnics/pnic/<pnic_name>/member_of /api/config/pnics/pnic/<pnic_name>/bond_mode /api/config/pnics/pnic/<pnic_name>/trunks	pnic <port_channel_name> type port_channel pnic <pnic_name> member_of <portchannel_name> show port-channel

Physical Network Interface Cards

Configuring LLDP

Starting from NFVIS 3.7.1 release LLDP is supported on NFVIS. The Link Layer Discovery Protocol (LLDP) is used by network devices for advertising their identity, capabilities, and neighbors. You can configure LLDP on a PNIC which is not a port channel or a DPDK port. By default, LLDP is disabled for all PNICs.

LLDP information is sent by devices from each of the interface at a fixed interval, in the form of an ethernet frame. Each frame contains one LLDP Data Unit (LLDPDU). Each LLDPDU is a sequence of type-length-value (TLV) structures.

LLDP is enabled in transmit and receive mode. The LLDP agent can transmit the local system capabilities and status information and receive the remote system's capabilities and status information

If LLDP is enabled on two connected devices, they can see each other as neighbors.

Note LLDP packets are not propagated to VMs. LLDP cannot be enabled on port channel or DPDK ports.

To enable LLDP on a PNIC:

```
configure terminal
pnic eth0 lldp enabled
commit
```

To disable LLDP on a PNIC:

```
configure terminal
pnic eth0 lldp disabled
commit
```

Use the **show lldp neighbors** command to display the peer information:

```
nfvis# show lldp neighbors eth0
----------------------------------------------------------------
DEVICE
NAME ID      HOLDTIME  CAPS  PLATFORM  PORTID  DESCRIPTION

----------------------------------------------------------------
eth0 Switch1623 120 Bridge, Router Cisco IOS Software, Catalyst L3 Switch Software
(CAT3K_CAA-UNIVERSALK9-M), Version 15.0(1)EX3, RELEASE SOFTWARE (fc2)Ifname:
Gi1/0/4GigabitEthernet1/0/4
```

Use the **show lldp stats** command to display the tx and rx information:

```
nfvis# show lldp stats eth0
-------------------------------------------------------------------
TX      DISCARD  ERROR  RX      DISCARDED  UNREC
NAME   FRAMES  RX       RX     FRAMES    TLVS       TLVS   AGEOUTS
-------------------------------------------------------------------
eth0    23      0        0       19667     0         0         0
```

LLDP Configuration APIs and Commands

APIs	Commands
/api/config/pnics/pnic/<pnic_name>/lldp /api/operational/lldp/neighbors /api/operational/lldp/stats	pnic <pnic_name> lldp enabled pnic <pnic_name> lldp disabled show lldp neighbors <pnic_name> show lldp stats <pnic_name>

Configuring Administrative Status of a Port

Administrator status provides a mechanism for configuring the administrative status of a port. It can be set to up or down and the default setting is on.

To configure adminstatus on a pnic for a VM:

```
configure terminal
pnic GE0-1 admin status down
commit
```

Use the **show pnic** command to verify the admin status configuration. Use the **show pnic link_state** command to verify the admin state configuration.

```
nfvis# show pnic GE0-1 link_state
link_state down
```

Admin Status Configuration APIs and Commands

APIs	Commands
/api/config/pnics/pnic/<pnic_name>/adminstatus	pnic <pnic_name> adminstatus

Tracking Changes for a Port

Note This feature is supported only on ENCS 5400 starting from NFVIS 3.10.1 release.

In a virtual environment when the PNIC goes down there is no indication to the interfaces inside the VNFs. It is useful to track state changes of PNICs including switch ports to one or more VNF interfaces and accordingly bring down or up the vNICs. This feature brings the appropriate interfaces inside the VNF up or down based on the PNIC state changes. Most of the VNFs support this functionality.

Track state can also be configured for LAN-SRIOV. The LAN network is not physically connected to LAN-SRIOV. Switch ports are connected to an embedded switch on the LAN side. The switch has an int-LAN interface which is a 10G interface the VMs can connect to from the LAN network using VFs (virtual functions). Therefore, the VM is not directly connected to LAN-SRIOV.

Track state configuration on WAN-SRIOV is not needed, as there is a one to one connection between WAN-SRIOV and the VM.

Track state can be configured for monitored and un-monitored VMs. If a track state configuration is deleted, the PNIC or switch port state changes will not be notified to the vNICs or VFs.

The VM has to be first deployed before you can configure PNIC track state for the VM. VNFs or vNICs do not have to be attached to a bridge connected to the PNIC.

To configure track state on a pnic for a VM use the following commands: **pnic <pnic_name> track-state <vm_name> <vnic>** or **pnic <pnic_name> track-state <deploy_name.vm_grp_name> <vnic>**

```
configure terminal
pnic GE0-0 track-state ROUTER 0
end
```

To verify the track state configuration on the VM use the **show interface** or **ethtool** commands or the VM specific command that displays the interface link state.

In the following example, the vedge VM deployed and vNIC 0 is being tracked by GE0-1. The **if-oper-status** command shows the state of the vNIC being tracked by pNIC. When GE0-1 is down, **if-oper-status** also shows as down.

Track StateAPIs and Commands

Track StateAPIs	Track State Commands
• api/config/pnics/pnic/<pnic_name>/track-state	• pnic <pnic_name> track-state <vm_name> <vnic> • pnic <pnic_name> track-state <deploy_name.vm_grp_name> <vnic>

Speed, Duplex and Autonegotiation

NFVIS supports autonegotiation by default on all PNICs. Speed and duplex are set to *auto* mode to indicate autonegotiation is enabled.

Autonegotiation allows a PNIC to communicate with the device on the other end of the link to determine the optimal duplex mode and speed for the connection. Autonegotiation can be turned off by configuring speed and duplex. Supported ethernet speed is 10 Mbps, 100 Mbps, and 1G and 10G.

Duplex mode displays the data flow on the interface. Duplex mode on an interface can be full or half duplex. A half-duplex interface, can only transmit or receive data at any given time and a full-duplex interface can send and receive data simultaneously.

When autonegotiation is enabled on a port, it does not automatically determine the configuration of the port on the other side of the ethernet cable to match it. Autonegotiation only works if it is enabled on both sides of the link. If one side of a link has auto-negotiation enabled, and the other side of the link does not, then autonegotiation cannot determine the speed and duplex configurations of the other side. If autonegotiation is enabled on the other side of the link, the two devices decide together on the best speed and duplex mode. Each interface advertises the speed and duplex mode at which it can operate, and the best match is selected. Higher speed and full duplex is the preferred mode.

If one side of a link does not have autonegotiation enabled, then the speed and duplex on both sides must match so that the data can transmit without collisions. Autonegotiation fails on 10/100 links, if one side of the link has been set to 100/full, and the other side has been set to autonegotiation which is 100/half.

Note Not all ports on ENCS 5000 series platform devices support auto-mdix feature. When autonegotiation is disabled you need to use the correct cable to configure speed and duplex correctly. The cable type depends on the remote system, based on which you can try straight through or cross over cable.

To disable autonegotiation on a PNIC, speed and duplex must be configured:

```
configure terminal
pnic GE0-0 speed 100 duplex full
commit
```

To enable autonegotiation on a PNIC:

```
configure terminal
pnic GE0-0 speed auto duplex auto
commit
```

To configure speed and duplex with non auto values:

```
configure terminal
pnic GE0-0 speed 100 duplex full
commit
```

Use the **show pnic GE0-0 operational-speed**, **show pnic GE0-0 operational-duplex** and **show pnic GE0-0 autoneg** to verify the configurations.

```
nfvis# show pnic GE0-0 operational-speed
operational-speed 100


nfvis# show pnic GE0-0 operational-duplex
operational-duplex full


nfvis# show pnic GE0-0 autoneg
autoneg off
```

To verify the PNIC speed and duplex configurations, use the **show notification stream nfvis Event** command.

```
notification
event Time 2019-12-16T22:52:49.238604+00:00
nfvisEvent
    user_id admin
    config_change true
    transaction_id 0
    status FAILURE
    status_code 0
    status_message Pnic GE0-1 speed did not update successfully
    details NA
    event_type PNIC_SPEED_UPDATE
    severity INFO
    host_name nfvis
    !
!
notification
event Time 2019-12-16T22:53:05.01598+00:00
nfvisEvent
    user_id admin
    config_change true
    transaction_id 0
    status SUCCESS
    status_code 0
    status_message Pnic GE0-1 duplex updated successfully:full
    details NA
    event_type PNIC_DUPLEX_UPDATE
    severity INFO
    host_name nfvis
    !
!
```

Speed, Duplex and Autonegotiation APIs and Commands

Speed, Duplex and Autonegotiation APIs	Speed, Duplex and Autonegotiation Commands
/api/config/pnics/pnic/GE0-0/speed	pnic GE0-0 speed auto duplex auto
/api/config/pnics/pnic/GE0-0/duplex	pnic GE0-0 speed 100 duplex full show
/api/operational/pnics/pnic/GE0-0/operational-speed	show pnic GE0-0 operational-speed
/api/operational/pnics/pnic/GE0-0/operational-duplex	show pnic GE0-0 operational-duplex
/api/operational/pnics/pnic/GE0-0/autoneg	show pnic GE0-0 autoneg

Dynamic SR-IOV

Dynamic Single-root input/output virtualization (SR-IOV) allows you to enable or disable SR-IOV on a Physical Network Interface Controller (PNIC). You can disable SR-IOV on any PNIC to 0 and enable SR-IOV by setting a value between 1 to maximum virtual functions (maxvfs) supported on PNICs. You can also create and delete SR-IOV networks based on the number of virtual functions (numvfs) set on that PNIC while enabling SR-IOV. Existing fresh installation behavior has not changed. Each PNIC has default number of VFs created and default SR-IOV networks are created. User can use CLI, API or GUI to enable or disable SR-IOV on a PNIC or to create or delete SR-IOV networks

Restrictions or Limitations

- The supported platforms are CSP-2100, CSP-5000, UCSc-C220-M5X and UCS-E-M3.

  Dynamic SR-IOV is not supported on ENCS 5000 series.

- Dynamic SR-IOV is not supported on certain PNICs:
  - PNIC with driver i40e

    Note PNIC with driver i40e is supported on default SR-IOV.

  - PNIC that does not support SR-IOV
- Only switch mode VEB is supported for NFVIS 3.12.1 release.
- Resizing the number of virtual functions is not supported. SR-IOV should be disabled and then enabled with desired number of virtual functions.

Disable SR-IOV on a PNIC

All SR-IOV networks for a PNIC must be deleted. PNIC should not be attached to a bridge.

```
configure terminal
no pnic eth0-1 sriov
commit
```

Enable SR-IOV on a PNIC

To enable SR-IOV on a PNIC, it has to support SR-IOV, numvfs field should be less than maximum supported VFs (maxvfs) on a PNIC and PNIC should not be attached to a bridge.

```
configure terminal
pnic eth0-1 sriov numvfs 20
commit
```

To display SR-IOV state of all PNICs use **show pnic sriov** command. To display SR-IOV state of individual PNIC use **show pnic eth0-1 sriov** command.

Creation of SR-IOV Networks

To create SR-IOV networks, PNIC must have SR-IOV enabled and configured with numvfs. The SRIOV network name must have the following format: <pnic_name>-SRIOV-<num> with <pnic_name> as a valid PNIC name and <num> must be greater than 0 and less than numvfs.

To create SR-IOV network in trunk mode:

```
configure terminal
networks network eth0-1-SRIOV-1 sriov true
commit
```

To create SR-IOV network in access mode:

```
configure terminal
networks network eth0-1-SRIOV-1 sriov true trunk false vlan 30
commit
```

Delete SR-IOV Networks

To delete SR-IOV networks VM should not be attached to the network.

```
configure terminal
no networks network eth0-1-SRIOV-1
commit
```

To verify the system networks use **show system networks** command.

System Routes

You can also configure static system routes along with the default routes in the system. Static routes are for traffic that should not go through the default gateway. When certain destinations are not reachable through the default routes, this configuration is effective. Also it updates the system routing table.

You can create a route by providing the destination and prefix length, but a valid route requires a specified device, a gateway or both. The gateway input represents the address of the nexthop router in the address family. The dev input is the name of the outbound interface for the static route.

Configuring System Routes

To configure additional system routes:

```
configure terminal
system routes route 172.25.222.024 gateway 172.25.221.1
```

```
system routes route 172.25.223.024 dev wan-br
commit
```

To verify the system routes configuration, use the **show system routes** command.

```
nfvis# show system routes

DESTINATION    PREFIXLEN       STATUS
----------------------------------------
172.25.222.0      24           Success
172.25.223.0      24           Success
```

System Routes APIs and Commands

System Routes APIs	System Routes Commands
/api/config/system/routes /api/config/system/routes/route/<host destination,netmask>	system routes route show system routes

Troubleshooting

To troubleshoot errors in configured routes, use **show system routes** command to identify the failed route. The following example shows common failures with system routes:

```
nfvis# show system routes

DESTINATION      PREFIXLEN       STATUS
-----------------------------------------
172.25.222.0        24          Failure(1)
172.25.223.1        24          Failure(2)
```

You can find the cause for each error in the *nfvos-confd* log.

```
Failure 1) result=RTNETLINK answers: Network is unreachable
```

In this failure *nfvos-confd* log indicates the network is unreachable. To resolve this issue you can either reconfigure the route with a reachable gateway or identify network connectivity issue.

```
Failure 2)result=RTNETLINK answers: Invalid argument
```

In this failure there is a mismatch between the subnet address and the prefix length. To resolve this issue you can reconfigure the route with the correct subnet address (in this case 172.25.223.0 for prefix length 24).

Cisco Network Plug-n-Play Support

Note Starting from 3.10.1 release, NFVIS is integrated with PnP 1.8.

The Cisco Network Plug and Play (Cisco Network PnP) solution provides a simple, secure, unified, and integrated offering for enterprise network customers to ease new branch or campus device rollouts, or for provisioning updates to an existing network. The solution provides a unified approach to provision enterprise

networks comprising Cisco routers, switches, and wireless devices with a near zero touch deployment experience. This solution uses Cisco Application Policy Infrastructure Controller Enterprise Module (APIC-EM) to centrally manage remote device deployments.

Currently, you can use the Cisco Network Plug and Play client to:

- Auto discover the server
- Provide device information to the server
- Bulk provisioning of user credentials

Bulk Provisioning of User Credentials

You can change the default user name and password of the devices using the Cisco Network PnP client. The Cisco Network PnP server sends the configuration file to Cisco Network PnP clients residing on multiple devices in the network, and the new configuration is automatically applied to all the devices.

Note For bulk provisioning of user credentials, ensure that you have the necessary configuration file uploaded to the Cisco APIC-EM. The following are the supported configuration formats:

Sample Format 1

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <rbac xmlns="http://www.cisco.com/nfv/rbac">
    <authentication>
      <users>
        <user>
          <name>admin</name>
          <password>Cisco123#</password>
          <role>administrators</role>
        </user>
        <user>
          <name>test1</name>
          <password>Test1239#</password>
          <role>administrators</role>
        </user>
        <user>
          <name>test2</name>
          <password>Test2985#</password>
          <role>operators</role>
        </user>
      </users>
    </authentication>
  </rbac>
</config>
```

Sample Format 2

If you use format 2, the system will internally convert this format into format 1.

```
<aaa xmlns="http://tail-f.com/ns/aaa/1.1">
    <authentication>
      <users>
        <user>
```

```
            <name>admin</name>
            <password>User123#</password>
           </user>
        </users>
    </authentication>
</aaa>
```

PnP Discovery Methods

When a device is powered on for the first time, the Cisco Network PnP agent discovery process, which is embedded in the device, wakes up in the absence of the startup configuration file, and discovers the IP address of the Cisco Network PnP server located in the Cisco APIC-EM. The Cisco Network PnP agent uses the following discovery methods:

- Static IP address—The IP address of the Cisco Network PnP server is specified using the **set pnp static ip-address** command.
- DHCP with option 43—The Cisco PnP agent automatically discovers the IP address of the Cisco Network PnP server specified in the DHCP option 43 string. For more details on how to configure DHCP for PnP server auto-discovery, see the Solution Guide for Cisco Network Plug and Play
- Domain Name System (DNS) lookup—If DHCP discovery fails to get the IP address of the PnP server, for example, because option 43 is not configured, the Cisco Plug and Play Agent falls back on a DNS lookup method. Based on the network domain name returned by the DHCP server, it constructs a fully qualified domain name (FQDN) for the PnP server, using the preset hostname "pnpserver". For more details on how to configure DNS for PnP server auto-discovery, see the Solution Guide for Cisco Network Plug and Play.

  Note DNS FQDN Only lookup method is supported since 3.10.1 release.

- Cloud Redirection—This method uses the Cisco Cloud Device Redirect tool available in the Cisco Software Central.The Cisco Plug and Play Agent falls back on the Cloud Redirection method if DNS lookup is not successful.

Configuring PnP Discovery Methods

To enable static mode for PnP discovery using IPv4:

```
configure terminal
pnp automatic dhcp disable dhcp-ipv6 disable dns disable dns-ipv6 disable cco disable
cco-ipv6 disable
pnp static ip-address 192.0.2.8 port 80 transport http
commit
pnp action command restart
```

To enable static mode for PnP discovery using IPv6:

```
configure terminal
pnp automatic dhcp disable dhcp-ipv6 disable dns disable dns-ipv6 disable cco disable
cco-ipv6 disable
```

```
pnp static ipv6-address 0:0:0:0:0:ffff:c000:208 port 80 transport http
commit
pnp action command restart
```

Note Either IPv4 or IPv6 can be enabled at a time.

To enable static mode for PnP discovery using FQDN:

```
configure terminal
pnp static ip-address apic-em-fqdn.cisco.com port 80 transport http
commit
```

Note In FQDN support for PnP, domain names can be specified as an input. FQDN that is configured with IPv6 on a DNS server is not supported.

To enable automatic mode for PnP discovery using IPv4:

Note By default, the automatic discovery mode for DHCP, DNS, and CCO is enabled. You can enable or disable the options as required. For example, you can enable all options or keep one enabled, and the rest disabled.

```
configure terminal
pnp automatic dhcp enable
pnp automatic dns enable
pnp automatic cco enable
pnp automatic timeout 100
commit
```

To enable automatic mode for PnP discovery using IPv6:

```
configure terminal
pnp automatic dhcp-ipv6 enable
pnp automatic dns-ipv6 enable
pnp automatic cco-ipv6 enable
pnp automatic timeout 30
commit
```

Note You cannot disable both static and automatic PnP discovery modes at the same time. You must restart PnP action every time you make changes to the PnP discovery configuration. You can do this using the **pnp action command restart**.

Verifying the PnP Status

Use the **show pnp** command in privileged EXEC mode to verify the configuration of PnP discovery methods. The following sample output shows that the static discovery mode is enabled, and the automatic discovery mode is disabled.

```
nfvis# show pnp
pnp status response "PnP Agent is running\n"
pnp status ip-address 192.0.2.8
pnp status ipv6-address ""
pnp status port 80
pnp status transport http
pnp status cafile ""
pnp status created_by user
pnp status dhcp_opt43 0
pnp status dns_discovery 0
pnp status cco_discovery 0
pnp status dhcp-ipv6 0
pnp status dns-ipv6 0
pnp status cco-ipv6 0
pnp status timeout 100
nfvis#
```

FQDN

```
nfvis# show pnp
pnp status response "PnP Agent is running\nserver-connection\n    status: Success\n    time:
 06:23:11 Jun 17\ndevice-info\n    status: Success\n    time: 06:23:06 Jun 17\nbackoff\n
  status: Success\n    time: 06:23:11 Jun 17\ncertificate-install\n    status: Success\n
  time: 06:21:38 Jun 17\ncli-exec\n    status: Success\n    time: 06:22:50 Jun 17\ntopology\n
    status: Success\n    time: 06:23:00 Jun 17\n"
pnp status ip-address apic-em-fqdn.cisco.com
pnp status ipv6-address ""
pnp status port 443
pnp status transport https
pnp status cafile /etc/pnp/certs/trustpoint/pnplabel
pnp status created_by user
pnp status dhcp_opt43 0
pnp status dns_discovery 0
pnp status cco_discovery 0
pnp status dhcp-ipv6 0
pnp status dns-ipv6 0
pnp status cco-ipv6 0
pnp status timeout 0
nfvis#
```

The following sample output shows that the static discovery mode is disabled, and the automatic discovery mode is enabled for DHCP, DNS, and CCO:

DHCP:

```
nfvis# show pnp
pnp status response "PnP Agent is running\nserver-connection\n    status: Success\n    time:
 05:05:59 Jun 17\ninterface-info\n    status: Success\n    time: 05:05:56 Jun
17\ndevice-info\n    status: Success\n    time: 05:05:38 Jun 17\nbackoff\n    status:
Success\n    time: 05:05:59 Jun 17\ncapability\n    status: Success\n    time: 05:05:44 Jun
 17\ncertificate-install\n    status: Success\n    time: 05:01:19 Jun 17\ncli-exec\n
status: Success\n    time: 04:58:29 Jun 17\ntopology\n    status: Success\n    time: 05:05:49
 Jun 17\n"
pnp status ip-address 192.0.2.8
pnp status ipv6-address ""
pnp status port 443
pnp status transport https
pnp status cafile /etc/pnp/certs/trustpoint/pnplabel
```

```
pnp status created_by dhcp_discovery
pnp status dhcp_opt43 1
pnp status dns_discovery 1
pnp status cco_discovery 1
pnp status dhcp-ipv6 1
pnp status dns-ipv6 1
pnp status cco-ipv6 1
pnp status timeout 60
```

DNS:

```
nfvis# show pnp
pnp status response "PnP Agent is running\nserver-connection\n    status: Success\n    time:
 05:13:55 Jun 17\ndevice-info\n    status: Success\n    time: 05:13:49 Jun 17\nbackoff\n
  status: Success\n    time: 05:13:55 Jun 17\ncertificate-install\n    status: Success\n
  time: 05:12:26 Jun 17\ncli-exec\n    status: Success\n    time: 05:13:34 Jun 17\ntopology\n
    status: Success\n    time: 05:13:45 Jun 17\n"
pnp status ip-address pnpserver.apic-em-fqdn.cisco.com
pnp status ipv6-address ""
pnp status port 443
pnp status transport https
pnp status cafile /etc/pnp/certs/trustpoint/pnplabel
pnp status created_by dns_discovery
pnp status dhcp_opt43 1
pnp status dns_discovery 1
pnp status cco_discovery 1
pnp status dhcp-ipv6 1
pnp status dns-ipv6 1
pnp status cco-ipv6 1
pnp status timeout 60
```

CCO:

```
nfvis# show pnp
pnp status response "PnP Agent is running\nserver-connection\n    status: Success\n    time:
 05:24:25 Jun 17\ninterface-info\n    status: Success\n    time: 05:23:13 Jun
17\ndevice-info\n    status: Success\n    time: 05:23:01 Jun 17\nbackoff\n    status:
Success\n    time: 05:24:25 Jun 17\ncapability\n    status: Success\n    time: 05:23:06 Jun
 17\nredirection\n    status: Success\n    time: 05:09:43 Jun 17\ncli-exec\n    status:
Success\n    time: 05:09:53 Jun 17\ncertificate-install\n    status: Success\n    time:
05:18:43 Jun 17\ntopology\n    status: Success\n    time: 05:23:10 Jun 17\n"
pnp status ip-address 192.0.2.8
pnp status ipv6-address ""
pnp status port 443
pnp status transport https
pnp status cafile /etc/pnp/certs/trustpoint/pnplabel
pnp status created_by cco_discovery
pnp status dhcp_opt43 1
pnp status dns_discovery 1
pnp status cco_discovery 1
pnp status dhcp-ipv6 1
pnp status dns-ipv6 1
pnp status cco-ipv6 1
pnp status timeout 60
```

PnP Server APIs and Commands

PnP Server APIs	PnP Server Commands
• /api/config/pnp • /api/config/pnp?deep	• pnp static ip-address • pnp automatic • show pnp

PnP Action

You can start, stop, and restart any PnP action using the PnP action command or API.

PnP Action API and Command

PnP Action API	PnP Action Command
• /api/operations/pnp/action	• pnp action command

DPDK Support on NFVIS

Data Plane Development Kit (DPDK) support on NFVIS is introduced to increase network throughput. DPDK allows applications to pull data directly from the Network Interface Card (NIC) without involving the kernel, therefore delivering high-performance user-space network I/O. DPDK support on NFVIS allows network traffic to bypass NFVIS kernel and directly reach deployed VNFs and service chains. For DPDK adption NFVIS reserves additional cores and memory to enhance system performance.

DPDK support on NFVIS was first introduced in NFVIS 3.10.1 release and enhancements were added in subsequent releases:

- NFVIS 3.10.1 – DPDK support only for service bridges. DPDK support can be enabled only when the device is in the factory default state. DPDK support is supported only on ENCS 5400 series devices.
- NFVIS 3.11.1 – DPDK support can be enabled at any time. All Virtual NICs connected to service bridges for all VNFs are upgraded to DPDK support. DPDK support is supported only on ENCS 5400 series devices.
- NFVIS 3.12.1 – DPDK support is extended to all supported platforms. Physical NICs can also use DPDK.

DPDK support on NFVIS includes:

- Upgrading existing bridges to enable DPDK
- Upgrading virtual NICs attached to VNFs to enable DPDK
- Upgrading physical NICs to enable DPDK

Note NICs and WAN side are not upgraded as they are configured with SR-IOV.

Once DPDK support is successfully enabled, you can disable DPDK only by resetting NFVIS to factory settings.

Restrictions

- SR-IOV interfaces and DPDK support:

Note This restriction does not apply to ENCS 5000 series devices.

To enable DPDK, every device driver must be supported by DPDK. NFVIS does not support SR-IOV interface upgrade to enable DPDK because SR-IOV device drivers are not supported by DPDK. If any SR-IOV network has been configured on an interface, that interface will not support DPDK. Also if an SR-IOV interface is attached to a bridge, the bridge does not support DPDK and if a bridge is supporting DPDK, any SR-IOV interface cannot be attached to it.

- VNF downtime:

  When DPDK support is enabled on a system, NFVIS upgrades virtual NICs attached to the VNFs. The VNFs are powered down causing a downtime for the VNF service for a short duration of time. After the upgrade is complete, all VNFs are powered up again.

System Requirements

DPDK support optimizes the performance by utilizing additional resources such as CPU and memory. If NFVIS is not able to acquire additional processing or memory, DPDK support can not be enabled.

Enabling DPDK support requires additional core from each socket available in the system. Depending upon the number of sockets present in the system, NFVIS acquires additional core for DPDK support.

Table 3: CPU Allocation

Total Cores	Before NFVIS 3.12.x	NFVIS 3.12.x Without DPDK support	NFVIS 3.12 with DPDK support
12 or less	1	1	1 + (1 core per socket)
Between 12 and 16 (including 16)	2	1	1 + (1 core per socket)
More than 16	4	2	2 + (1 core per socket)

Note If hyper-threading is enabled on the device, each core reflects two vCPUs in NFVIS portal under system resource allocation.

The amount of memory required for DPDK support is summarized in the table below.

Table 4: Memory Allocation

Total System Memory	Reserved for NFVIS	Additional memory required for DPDK support
Up to 16 GB	3 GB	1 GB
Up to 32 GB	3 GB	1 GB
Up to 64 GB	4 GB	2 GB

Total System Memory	Reserved for NFVIS	Additional memory required for DPDK support
Up to 128 GB	4 GB	4 GB

Note The additional memory required for DPDK support is counted per NUMA node available on the system.

Configuring DPDK Support on NFVIS

Configuring DPDK support takes up to a minute and network changes can be observed during the process. NFVIS provides an operational status for DPDK support which indicates if DPDK support is enabled or not. The different values for operational status are listed in the table below.

DPDK Status	Description
disabled	The system is not using DPDK.
enabled	DPDK support is successfully enabled on the system. Additional CPU and memory resources are reserved for DPDK.
enabling	The system is in the process of enabling DPDK.
error	The system is unable to acquire the required resources to support DPDK. All of the resources that were acquired by DPDK are released again.

If DPDK status is in error state, DPDK support can be manually disabled. Before enabling DPDK again, reboot the system to defragment the system memory and increase the chance of resource allocation for a successfull configuration.

After enabling DPDK, SR-IOV configured physical NICs will not be able to interact with DPDK bridges. To add a physical NIC to a DPDK bridge, all SR-IOV networks created on the interface should be removed first. NFVIS will not allow adding an SR-IOV configured interface to a DPDK bridge. For more information, see <dynamic sriov link>

To enable DPDK support:

```
config terminal
system setting dpdk enable
commit
```

To display the operational status that indicates DPDK support, use **show system native settings** command.

```
nfvis# show system settings-native dpdk-status
system settings-native dpdk-status enabled
```

If NFVIS is unable to acquire sufficient resources, it shows an error state, and DPDK configuration can be removed. After removing the configuration, DPDK can be enabled again.

```
nfvis# show system settings-native dpdk-status
system settings-native dpdk-status error
```

```
config terminal
no system settings dpdk
commit

nfvis# show system setting-native dpdk-status
system settings-native dpdk-status disabled
```

Storage Access

-

Network File System Support

The Network File System (NFS) is an application where the user can view, store and update the files on a remote device. NFS allows the user to mount all or a part of a file system on a server. NFS uses Remote Procedure Calls (RPC) to route requests between the users and servers.

NFS Mount and Unmount

To mount NFS:

```
configure terminal
system storage nfs_storage
nfs
100
10.29.173.131
/export/vm/amol
commit
```

To unmount NFS use **no system storage nfs_storage** command.

Image Registration on NFS

Images in tar.gz, ISO and qcow2 format, remote images and images on mounted NFS can be registered on NFS.

To register tar.gz images on NFS:

```
configure terminal
vm_lifecycle images image myas10 src file:///data/mount/nfs_storage/repository/asav961.tar.gz
properties property placement value nfs_storage
commit
```

Similar configuration can be used for the various images formats.

To unregister an image from NFS use **no vm_lifecycle images** command.

Deploy VM on NFS

To deploy a VM on NFS, under deployment vm group use **placement type zone_host host nfs_storage** command.

External Storage for Cisco ENCS 5400

For details on supported storage type, number of storage devices and, RAID modes on each hardware, see the table below:

Device	Storage Details
ENCS 5400	Cisco 5400 Enterprise Network Compute System Hardware Installation Guide

Note RAID controller is optional on ENCS 5400.

RAID configurations are performed from Cisco IMC for each hardware platform. For UCS-E devices, all RAID configurations should be performed before NFVIS installation. For ENCS 5400, RAID configurations can be done even after the NFVIS is installed, as the installation is not done on an external storage.

For each hardware platform a maximum of two external disks are supported. Starting from NFVIS 3.8.1 release, external disks are supported on ENCS 5400. For ENCS 5400, if the external disk is raided into a single virtual group, it shows up as extdatastore1. Without the RAID card, ENCS 5400 can support multiple external disks called as extdatastore1 and extdatastore2 depending upon the slot it occupies.

Note Power off the system before you remove or insert disks in ENCS 5400.

To display the number of external disks on the system, use the **show system ext-disks** command.

```
nfvis# show system ext-disks

NAME
---------------
extdatastore1
```

To display the disk space on an external disk, use the **show system disk-space** command.

```
nfvis# show system disk-space

ASSOCIATED
          PHYSICAL    TOTAL  SIZE  SIZE       USE
DISK NAME DISK        SIZE   USED  AVAILABLE  PERCENT
-------------------------------------------------------------
lv_data         sde2        99G    4.3G  94G        5%
lv_var          sde2        3.9G   245M  3.4G       7%
lv_root         sde2        7.8G   1.9G  5.5G       26%
extdatastore1   sda         917G   77M   871G       1%
```

Host System Operations

This section describes operations that can be performed on the NFVIS host.

Power Cycle System

To power cycle NFVIS, use the following command:

```
nfvis# hostaction powercycle
```

A notification and syslog is sent to indicate that a power cycle was performed.

Reboot System

To reboot NFVIS, use the following command:

```
nfvis# hostaction reboot
```

A notification and syslog is sent to indicate the system reboot.

Shutdown System

To shutdown NFVIS, use the following command:

```
nfvis# hostaction shutdown
```

A notification and syslog will be sent to indicate that the system was shutdown.

System file-list

To view a list of files on the system, use the **show system file-list** command.

```
nfvis# show system file-list [disk [local | nfs | usb] ]
```

Disk Type	Files
local	Files present in the internal datastore and external datastores
nfs	Files on NFS
usb	Files on the mounted USB drive

System file-copy

To copy a file from the USB drive to the /data/intdatastore/uploads directory, use the **system file-copy** command. To copy a VM image from the USB drive:

```
configure terminal
system usb-mount mount active
system file-copy usb file name usb1/package/isrv-universalk9.16.03.01.tar.gz
commit
```

The **system file-copy** command can also be used to copy a file from the given source path to the given destination path. The allowed directories for source path and destination path are /data/intdatastore, /mnt/extdatastore1, /mnt/extdatastore2 and /data/mount.

```
nfvis# system file-copy source <path-to-source-file> destination <path-to-destination-file>
```

System file-delete

The **system file-delete** command is used to delete a file from one of these directories: /data/intdatastore, /mnt/extdatastore1, /mnt/extdatastore2, /mnt-usb/ or /data/mount

```
nfvis# system file-delete file name
/data/intdatastore/uploads/isrv-universalk9.16.03.01.tar.gz
```

Secure Copy

The secure copy (**scp**) command allows only the admin user to securely copy files from NFVIS to an external system, or from an external system to NFVIS. For example, this command can be used to copy an upgrade package to NFVIS.

The syntax for this command is:

scp *<source> <destination>*

Note For detailed information about how to use the **scp** command to copy to or from supported locations, see the **scp** section in Cisco Enterprise Network Function Virtualization Infrastructure Software Command Reference.

SCP between two NFVIS devices is not supported.

Examples

The following example copies the sample.txt file from intdatastore to an external system.

```
nfvis# scp intdatastore:sample.txt user@203.0.113.2:/Users/user/Desktop/sample.txt
```

The following example copies the test.txt file from an external system to intdatastore.

```
nfvis# scp user@203.0.113.2:/Users/user/Desktop/test.txt intdatastore:test_file.txt
```

The following example copies the test.txt file from an external system to USB.

```
nfvis# scp user@203.0.113.2:/user/Desktop/my_test.txt usb:usb1/test.txt
```

The following example copies the sample.txt file to an NFS location.

```
nfvis# scp user@203.0.113.2:/user/Desktop/sample.txt nfs:nfs_test/sample.txt
```

The following example copies the sample.txt file from an external system with IPv6 address.

```
nfvis# scp user@[2001:DB8:0:ABCD::1]:/user/Desktop/sample.txt intdatastore:sample.txt
```

The following example copies the nfvis_scp.log file to an external system.

```
nfvis# scp logs:nfvis_scp.log user@203.0.113.2:/Users/user/Desktop/copied_nfvis_scp.log
```

The following example shows how to secure copy from techsupport as source:

```
nfvis# scp logs:nfvis_techsupport.tar.gz
user@203.0.113.2:/Users/user/Desktop/copied_techsupport.tar.gz
```

Change BIOS Password

This command is applicable only to the ENCS platform. It allows the user to change the BIOS password. A notification and syslog are sent regarding the password change.

To change the BIOS password:

```
nfvis# hostaction change-bios-password <new-password>
```

There is a strong password check enforced for the new BIOS password. The new password should contain:

- At least one lowercase character
- At least one uppercase character
- At least one number
- At least one special character from #, @ or _
- Password length should be between 7 and 20 characters
- The first character cannot be a #

Change CIMC Password

This command is applicable only to the ENCS platform. It allows the user to change the CIMC password. A notification and syslog are sent regarding the password change.

To change CIMC password:

```
nfvis# hostaction change-cimc-password <new-password>
```

There is a strong password check enforced for the new CIMC password. The new password should contain:

- At least one lowercase character
- At least one uppercase character
- At least one number
- At least one special character from #, @ or _
- Password length should be between 8 and 20 characters

Route Distribution

The Route Distribution feature works together with a remote BGP router. It allows you to announce or withdraw specified routes to the remote BGP router.

You can use this feature to announce the route of int-mgmt-net subnet to a remote BGP router. A remote user, can access the VMs attached to int-mgmt-net through the VMs' IP address on int-mgmt-net-br through a BGP router, when the routes are successfully inserted on the remote BGP router.

To configure or update route distribution:

```
configure terminal
route-distribute 172.25.221.17local-bridge wan-br local-as 45.45remote-as 65000 network-subnet
```

```
 12.12.12.0/24
commit
```

Table 5: Property Description

Property	Type	Description	Mandatory
neighbor-address	IPv4	BGP neighbor IPv4 address. It is the key of the route distribution list.	Yes
local-address	IPv4	Local IPv4 address. This address must be configured as neighbor IP address on the remote BGP router. If not configured, local-address is set to local-bridge's IP address.	No
local-as		Local autonomous system number. It can be in following two formats: <decimal number, 1.0 .. 65535.65535><unsignedInt, 1 .. 4294967295>	Yes
local-bridge		Local bridge name for advertising routes (default wan-br).	No
remote-as		Remote autonomous system number. It can be in following two formats: <decimal number, 1.0 .. 65535.65535><unsignedInt, 1 .. 4294967295>	Yes
router-id	IPv4	Local router ID	No
network-subnet		List of network subnet to be announced.	Yes
subnet	IPv4 prefix	Network subnet to be announced H.H.H.H/N	Yes
next-hop	IPv4	IPv4 address of next hop. Default local-address or IP address of local-bridge.	No

Use the **no route-distribute** command to delete route distribution. To verify the route-distrubition status use the **show route-distribution** command.

Remote BGP Router Configuration Example

The NFVIS route distribution feature works together with the remote BGP router. The configuration on NFVIS and on remote BGP router must match.

This example shows the configuration on a remote BGP router.

```
router bgp 65000
  bgp log-neighbor-changes
  neighbor 172.25.221.106 remote-as 45.45
  neighbor 172.25.221.106 update-source GigabitEthernet2
```

Backup and Restore NFVIS and VM Configurations

Starting from NFVIS 3.10.1 release, you can backup and restore NFVIS configurations and VMs. You can also restore a backup from one NFVIS device to another if they are running on the same version of NFVIS and have the same platform.

Note To backup or restore a single VM, use vmImportAction and vmBackupAction APIs.

Restrictions for Backup and Restore on NFVIS

- The backup includes all deployed VMs except the registered images and uploaded files.
- VM backup failure results in failure of the whole system backup process.
- VM restore (including *hostaction restore* and *vmImportAction*) requires original registered image on the system, on the same datastore. Missing registered image or image registered in a different datastore results in VM restore failure.
- NFVIS VM backup does not support differential disk backup and every backup is a full VM backup.
- In case of multiple deployments based on a single registered image, every VM backup includes the registered image disk.
- The time taken to backup a VM depends on the option you choose:
  - *configuration-only* - within 1 min.
  - *configuration-and-vms* - depends on the number of VM deployments on your system, system disk write speed, and compress the VM disks into one bundle.
- The BACKUP_SUCCESS notification implies that the backup process has started successfully and does not indicate a successful system backup.
- Backup of a large deployment is time consuming and can result in failure due to insufficient disk space. The backup process cleans up the temporary files if the disk space is insufficient.
- You can either backup all the VMs or none.
- The final backup is a compressed file which requires temporary disk space to create the VM backup file. If the system has only one datastore, the maximum deployment backups in a single file is around one-third

to half of the datastore disk space. If the deployments occupies more disk space, use *vmExportAction* to backup an individual VM instead of relying on host backup for all VM deployments.

Backup and Restore

To backup and save NFVIS and all VM configurations use **configuration-only** option. To backup and save VM disks, NFVIS and VM configurations use **configuration-and-vms** option.

You can only create a backup to datastore or uploads directory, mounted USB device, or NFS mounted datastore. Without specifying, the backup file will have *.bkup* extension.

The following examples shows the backup options:

```
nfvis# hostaction backup configuration-and-vms file-path intdatastore:sample
```

```
nfvis# hostaction backup configuration-only file-path extdatastore2:sample-dir/sample
```

The following example shows the backup stored on a USB:

```
nfvis# hostaction backup configuration-only file-path usb:usb1/sample
```

Use the **hostaction backup force-stop** command to stop the running backup.

To restore a previous backup on an existing NFVIS setup or on a new NFVIS setup use except-connectivity option which preserves connectivity of the NFVIS and restores everything else from backup.

Note In hostaction restore process, the full file name (with *.bkup* extension) is required in the CLI.

```
nfvis# hostaction restore file-path intdatastore:sample.bkup
```

The following example shows how to restore a backup on a different NFVIS device:

```
nfvis# hostaction restore except-connectivity file-path extdatastore2:sample-dir/sample.bkup
```

Backup, Restore and Factory-Default-Reset

To perform **hostaction backup** -> **factory-default-reset** -> **hostaction restore** on the same box without any external storage (like USB or NFS mount), check the following issues:

- Backup file location:
  - The system backup bundle is saved under */datastore/uploads/* by default.

- Factory-default-reset cleans up all files under */datastore/uploads/*, but leave files under */datastore/* intact.
- **hostaction restore** requires backup bundle saved under */datastore/uploads/*. The restore process will not start if the backup bundle is saved in another location (bundle saved on USB or NFS should be copied to *datastore/uploads/ folder*).

- System requirements if system backup bundle contains VM backups:
  - VM restoration requires the original image or template registered in NFVIS.
  - Factory-default-reset all clean ups all registered images and uploaded files. You need to configure minimum setup, like host connection and upload registered images to the same datastore.

To prevent backup bundle from deleting with factory-default-reset:

- Save the backup bundle in remote locations. Then restore the connectivity and upload the backup bundle after reset.
- Save backup bundle in local */datastore/* and not in */datastore/uploads/* or copy backup bundle from */datastore/uploads/* to */datastore/*:

```
# Backup & Restore on the same NFVIS box without NFS & USB
# [[ BACKUP ]]
# before executing factory-default-reset

nfvis# nfvis# hostaction backup configuration-only file-path
extdatastore1:configBackup-01.bkup
nfvis# system file-copy source /mnt/extdatastore1/uploads/configBackup-01.bkup destination
 /mnt/extdatastore2/

# after factory-default-reset all-except-images or all-except-images-connnectivity,
# file /mnt/extdatastore1/uploads/configBackup-01.bkup will be deleted
# but /mnt/extdatastore2/configBackup-01.bkup won't.

# [[RESTORE]]
# after NFVIS rebooted and login to console, copy file to uploads/ directory

nfvis# system file-copy source /mnt/extdatastore2/configBackup-01.bkup destination
/mnt/extdatastore2/uploads/
nfvis# hostaction restore file-path extdatastore2:configBackup-01.bkup
```

For VM restoration:

- Use **factory-default-reset all-except-images** or **factory-default-reset all-except-images-connectivity** command to keep original registered images intact.
- If you use **factory-default-reset all** command, you need to upload and register images before running any **hostaction restore** action.

APC UPS Support and Monitoring

Note This feature is supported only on ENCS 5400.

This feature provides support for monitoring battery status for an APC UPS connected to the ENCS box through a USB cable. NFVIS gracefully shuts down when the UPS battery reaches 5% and boots up again when the battery reaches 15%. This feature is available only through NFVIS CLI and is disabled by default.

Incase of a prolonged power outage that drains the UPS battery completely, the box is powered off. When power is restored to the UPS, CIMC boots up which in turn boots up the NFVIS.

To enable APC UPS support feature:

```
apcups enable
```

To disable APC UPS support feature:

```
apcups disable
```

To check the battery status of an APC UPS:

```
apcups battery-status
```

Resetting to Factory Default

Factory default reset is available on all NFVIS supported hardware platforms.

You can reset the host server to factory default with the following three options :

- Reset all—Deletes VMs and volumes, files including logs, images, and certificates. Erases all configuration. Connectivity will be lost, and the admin password will be changed to factory default password..
- Reset all-except-images—Delete VMs and volumes, files including logs, user uploaded files and certificates. Erases all configuration except registered images. Connectivity will be lost, and the admin password will be changed to factory default password..
- Reset all-except-images-connectivity—Deletes VMs and volumes, files including logs and certificates. Erases all configuration except images, network, and connectivity.

Note Factory default rese tmust be used only for troubleshooting purpose. We recommend you contact Cisco Technical Support before performing factory default reset. This feature will reboot the system. Do not perform any operations for at least twenty minutes until the system reboots successfully.

To execute factory default reset:

```
nfvis#factory-default-resetall|all-except-images|all-except-images-connectivity
```

Note Enter **Yes** when you are prompted with the factory default warning message or **no** to cancel.

Factory Default APIs and Commands

Factory Default APIs	Factory Default Commands
• /api/operations/factory-default-reset/all • /api/operations/factory-default-reset/all-except-images • /api/operations/factory-default-reset /all-except-images-connectivity	• factory-default-reset

Configure Banner, Message of the day and System Time

Configuring Your Banner and Message of the Day

Cisco Enterprise NFVIS supports two types of banners: system-defined and user-defined banners. You cannot edit or delete the system-defined banner, which provides copyright information about the application. Banners are displayed on the login page of the portal.

You can post messages using the Message of the Day option. The message is displayed on the portal's home page when you log into the portal.

To configure your banner and message:

```
configure terminal
banner-motd banner "This is a banner" motd "This is the message of the day"
commit
```

Note Currently, you can create banners and messages in English only. You can view the system-defined banner using the **show banner-motd** command. This command does not display the user-defined banner or message.

Banner and Message APIs and Commands

Banner and Message APIs	Banner and Message Commands
• /api/config/banner-motd • /api/operational/banner-motd	• banner-motd • show banner-motd

Setting the System Time Manually or With NTP

You can configure the Cisco Enterprise NFVIS system time manually or synchronise with an external time server using Network Time Protocol (NTP).

To set the system time manually:

```
configure terminal
system set-manual-time 2017-01-01T00:00:00
commit
```

Note NTP is automatically disabled when the time clock is set manually.

To set the system time using NTP IPv4:

```
configure terminal
system time  ntp preferred_server 209.165.201.20 backup_server 1.ntp.esl.cisco.com
commit
```

To set the system time using NTP IPv6:

```
configure terminal
system time  ntp-ipv6 2001:420:30d:201:ffff:ffff:fff4:35
commit
```

Verifying the System Time Configuration

To verify all system time configuration details, use the **show system time** command in privileged EXEC mode as shown below:

```
nfvis# show system time

system time current-time 2017-01-01T17:35:39+00:00

system time current-timezone "UTC (UTC, +0000)"

REMOTE               REFID   ST    T         WHEN       POLL       REACH         DELAY
     OFFSET        JITTER
==============================================================================

*calo-timeserver     .GPS.    1          u           4   64           1         69.423
     2749736        0.000

* sys.peer and synced, o pps.peer, # selected, + candidate,

- outlyer, . excess, x falseticker, space reject
```

If the NTP server is invalid, it will not be displayed in the table. Also, when an NTP server is queried, if a response is not received before the timeout, the NTP server will also not be displayed in the table.

System Time APIs and Commands

APIs	Commands
• /api/operations/system/set-manual-time • /api/config/system/time/ntp/preferred_server • /api/config/system/time/ntp/backup_server • /api/config/system/time/timezone • /api/operational/system/time?deep	• system time • show system time • system set-manual-time

Configuring System Logs

NFVIS generates log files for troubleshooting issues. The configuration log and the operational log are the two main system log files. The configuration log has information related to configurations and actions performed on the system such as creation of networks. The operational log has information related to system operation such as statistics collection and monitoring.

Log entries can be one of the following types:

Log Level	Purpose
DEBUG	Information, typically of interest only when diagnosing problems.
INFO	Confirmation that things are working as expected.
WARNING	An indication that something unexpected happened, or indicative of some problem in the near future (for example, ‘disk space low’). The software application is still working as expected.
ERROR	Due to a serious problem, the software application is not able to perform some function.
CRITICAL	A serious error, indicating that the program itself may not be able to continue running.

By default, the configuration log has a log-level of INFO. All logs of type INFO, WARNING, ERROR and CRITICAL are logged.

By default, the operational log has a log-level of WARNING. All logs of type WARNING, ERROR and CRITICAL are logged.

The log-level for these log files can be changed using the **system set-log** command:

```
system set-log level error logtype configuration
```

The change to the log level is not persistant across a reboot. After a reboot, the default log levels are used.

The current log files are kept in the */var/log* directory in the system:

- show log - To display the list of available log files
- show log {filename} - To display the contents of a specific log file

Log Rotation

There is a size limit for the log files, under */var/log/* directory. When the log files reach the size limit, the location of logs is rotated to another place. The space limit for the total size of all rotated log files is 2 GB. The older log files are dropped automatically on reaching the space limit. You can also execute a command to trigger the log rotation procedure. The log files are monitored periordically and if a log file gets too big, it is rotated to another place.

There is a size limit for the log files stored in the */var/log* directory. The size of the log files is monitored periodically every fifteen minutes and if a log file gets too big, it is rotated to the */data/intdatastore/logs* directory. The space limit for the total size of all the rotated log files is 2 GB. The older log files are dropped automatically on reaching the space limit. You can also execute the **logrotate** command to trigger the log rotation procedure.

```
nfvis# logrotate
```

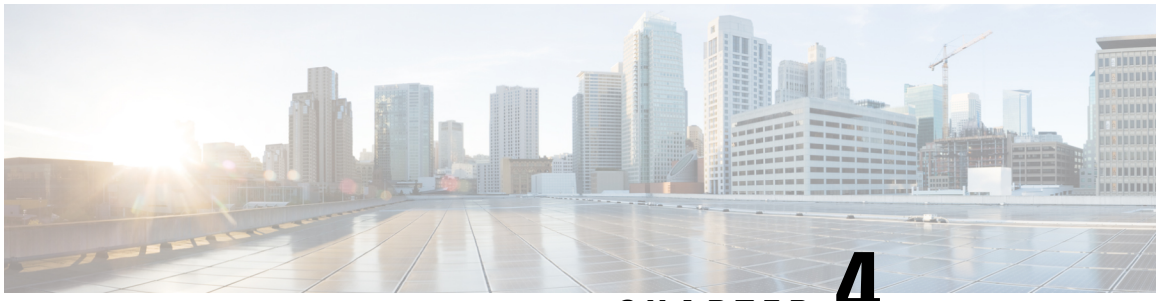
Verifying the System Log Configuration

To verify the system log configuration, use the **show system logging-level** command as shown below:

```
nfvis# show system logging-level
system logging-level configuration error
system logging-level operational warning
```

System Log APIs and Commands

System Log APIs	System Log Commands
• /api/operations/system/set-log • /api/operational/system/logging-level	• system set-log logtype [all/configuration/operational] level [critical/debug/error/info/warning] • show system logging-level



CHAPTER 4

VM Life Cycle Management

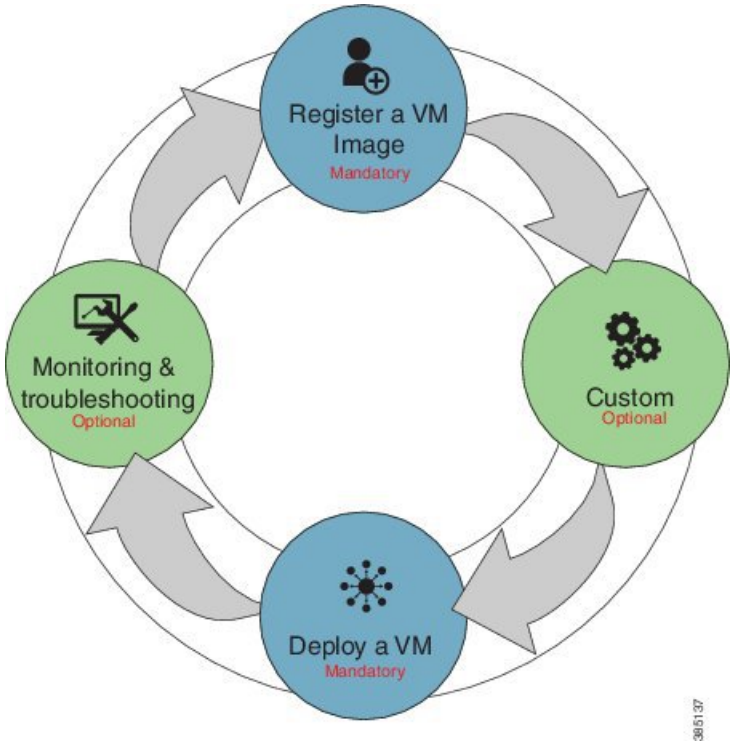
Overview of VM Life Cycle Management

VM life cycle management refers to the entire process of registering, deploying, updating, monitoring VMs, and getting them service chained as per your requirements. You can perform these tasks and more using a set of REST APIs or NETCONF commands or the Cisco Enterprise NFVIS portal.

Workflow of VM Life Cycle Management

The following diagram depicts the basic workflow of the VM life cycle management using REST APIs:

Figure 1: VM Life Cycle Management



1. **Register a VM Image—**To register a VM image, you must first copy or download the relevant VM image to the NFVIS server, or host the image on a http or https server. Once you have downloaded the file, you can register the image using the registration API. The registration API allows you to specify the file path to the location (on the http/https server) where the tar.gz file is hosted. Registering the image is a one-time activity. Once an image is registered on the http or https server, and is in active state, you can perform multiple VM deployments using the registered image.

2. **Customizing the VM—**After registering a VM image, you can optionally create a custom profile or flavor for the VM image if the profiles defined in the image file do not match your requirement. The flavor creation option lets you provide specific profiling details for a VM image, such as the virtual CPU on which the VM will run, and the amount of virtual memory the VM will consume.

   Depending on the topology requirement, you can create additional networks and bridges to attach the VM to during deployment.

3. **Deploy a VM—** A VM can be deployed using the deployment API. The deployment API allows you to provide values to the parameters that are passed to the system during deployment. Depending on the VM you are deploying, some parameters are mandatory and others optional.

4. **Manage and Monitor a VM—**You can monitor a VM using APIs and commands that enable you to get the VM status and debug logs. Using VM management APIs, you can start, stop, or reboot a VM, and view statistics for a VM such as CPU usage.

   A VM can also be managed by changing or updating its profile. You can change a VM's profile to one of the existing profiles in the image file; alternatively, you can create a new custom profile for the VM.

   The vNICs on a deployed VM can also be added or updated.

Note Before performing the VM life cycle management tasks, you will have to upload the VM images to the NFVIS server or http/s server.

For details on APIs, see the VM Lifecycle Management APIs chapter in the *API Reference for Cisco Enterprise Network Function Virtualization Infrastructure Software*.

Uploading VM Images to an NFVIS Server

You can upload VM images to an NFVIS server in the following ways. The files are copied to the default location (/data/intdatastore/uploads) on the host server.

- Copy the images from your local system to the NFVIS server—Use the **Image Upload** option from the Cisco Enterprise NFVIS portal.
- Copy the images using the USB drive—Ensure that you have plugged the USB drive that contains the required images into the server before mounting the USB drive.
- Copy using the **scp** command (scp username@external_server:/path/image.tar.gz intdatastore:image.tar.gz).

To copy an image using the USB device:

```
configure terminal
system usb-mount mount ACTIVE
system file-copy usb file name usb1/package/isrv-universalk9.16.03.01.tar.gz
commit
```

Note Use the **show system file-list disk usb** command in privileged EXEC mode to view a list of files available with the mounted USB drive. To save space, you can delete all unwanted text and TAR files from the default location using the **system file-delete** command in global configuration mode.

Verifying the Image Copied from the USB Drive

After copying the file from the USB drive to the host server, you can verify the file using the **show system file-list disk local** command:

```
nfvis# show system file-list disk local

SI NO     NAME     PATH                                                SIZE  TYPE    DATE MODIFIED
-------------------------------------------------------------------------------------------------------
1 lastlog-20170314.gz /data/intdatastore/logs/2017-03/14/10-00 337 Other 2017-03-14 21:55:42

2 escmanager-tagged-log.log-20170314.gz /data/intdatastore/logs/2017-03/14/10-00 167K Other
 2017-01-18 05:58:26
3 confd_audit.log-20170317.gz /data/intdatastore/logs/2017-03/17/09-30 4.6K Other 2017-03-17
 21:29:59
4 esc_postinit.log-20170317.gz /data/intdatastore/logs/2017-03/17/05-00 605K Other 2017-03-17
 16:40:19
5 error.log-20170317.gz /data/intdatastore/logs/2017-03/17/05-00 1.3K Other 2017-03-17
16:40:15
6 ovs-ctl.log-20170317.gz /data/intdatastore/logs/2017-03/17/12-00 20 Other 2017-03-16
```

```
00:00:01 4:01
!
!
!
62 ovs-ctl.log-20170323.gz /data/intdatastore/logs/2017-03/23/12-00 20 Other 2017-03-22
00:00:01
63 CentOS-7-x86_64-Everything-1511.ova /data/intdatastore/uploads 1.1G VM 2017-03-15 19:20:03
 Package
64 TinyLinux.tar.gz /data/intdatastore/uploads 17M VM 2017-03-15 18:25:00 Package
65 Cisco-KVM-vWAAS-1300-6.3.0-b98.tar.gz /data/intdatastore/uploads 979M VM 2017-03-15
19:19:11 Package
66 ubuntu_14.04.3-server-amd64-disk1.tar /data/intdatastore/uploads 527M VM 2017-03-15
19:20:17.gz Package
67 asav961.tar.gz /data/intdatastore/uploads 164M VM 2017-03-15 18:24:57 Package
68 isrv-universalk9.16.03.01.tar.gz /data/intdatastore/uploads 1.3G VM 2017-03-15 19:19:53
```

Related APIs and Commands

APIs	CLI Commands
• /api/operations/system/file-copy/usb/file • /api/config/system/usb-mount	• system file-copy usb file name • system usb-mount mount ACTIVE • system file-delete • show system file-list disk usb • show system file-list disk local

Performing Resource Verification

Given below are the APIs and commands to perform different types of resource verification:

Task	API	Command
To display CPU information for each CPU or the user specified CPU, and the VMs pinned to the CPU	• /api/operational/resources/cpu-info/cpus • /api/operational/resources/cpu-info/cpus/cpu • /api/operational/resources /cpu-info/cpus/cpu/<cpu-id>	show resources cpu-info cpus
To display information on the VMs running in all the physical CPUs or a specific physical CPU in the system	• /api/operational/resources/cpu-info/vnfs • /api/operational/resources/cpu-info/vnfs/vnf • /api/operational/resources/cpu-info/vnfs/vnf/ <deployment_name>.<vm_group_name>	show resources cpu-info vnfs

Task	API	Command
To get information on the number of CPUs allocated to VMs and the CPUs that are already used by the VMs	/api/operational/resources/cpu-info/allocation	show resources cpu-info allocation

Note To display information on all CPUs, VMs pinned to the CPUs, and VMs allocated to the CPUs, use the **show resources cpu-info** command.

CPU Over-Subscription

Cisco Enterprise NFVIS does not allow CPU over-subscription for low-latency network appliance VMs (for example, Cisco ISRv and Cisco ASAv). However, the CPU over-subscription is allowed for non low-latency VMs (for example, Linux Server VM and Windows Server VM).

Configuring Management IP Subnet

By default, all VMs with management interfaces will be provisioned with an IP address in the subnet of 10.20.0.1. To change the default subnet, the following commands needs to be executed in a sequence to first delete an exiting subnet and then add a new subnet in the network. For these commands to work, ensure there is no managed VNF's in the system before you change management network address.

To delete an existing subnet use **no vm_lifecycle networks network int-mgmt-net subnet int-mgmt-net-subnet** command.

To create a new subnet:

```
configure terminal
vm_lifecycle networks network int-mgmt-net subnet int-mgmt-net-subnet address 105.20.0.0
gateway 105.20.0.1 netmask 255.255.255.0 dhcp false
commit
```

VM Image Packaging

VM Image Packaging is a tool for converting qcow2 and img images into a tar.gz format with additional properties and profiles. VM image packaging can be done in two ways:

- VM Image Packaging Utility: This is an enhanced packaging process that allows the VM owner to run the **nfvpt.py** utility as a command with a combination of parameters to package the VM.
- Standard Image Packaging: This is a manual process in which a raw disk image (qcow2, img) is packaged along with the image properties file and bootstrap files (if needed) into a TAR archive file.

VM Image Packaging Utility

A VM image package is a TAR archive file with the root disk image and other descriptor files. This packaging method simplifies the process of a VM image registration and deployment. The attributes specified for the image enable resource requirement specification, creation of VM profiles, and a host of other properties for the VM.

The Cisco Enterprise NFVIS VM image packaging tool, nfvpt.py, helps VM owners package their VMs. The tool takes one or more qcow2 images (raw disk file) as the input file along with VM specific properties, bootstrap configuration files (if any), and generates a compressed TAR file.

Contents

The VM image packaging utility contains the following:

- nfvpt.py—It is a python based packaging tool that bundles the VM raw disk image/s along with VM specific properties.
- image_properties_template.xml—This is the template file for the VM image properties file, and has the parameters with default values. If the user provides new values to these parameters while creating the VM package, the default values get replaced with the user-defined values.
- nfvis_vm_packaging_utility_examples.txt—This file contains examples on how to use the image packaging utility to package a VM image.

Usage

To get the list of parameters that can be included in the command, and to get an explanation of each of the parameters, run the **help** command for the tool.

```
nfvpt.py --help
optional arguments:
  -h, --help            show this help message and exit
  --json JSON           Provide JSON input for bootstrap variables; mutually
                        exclusive with custom and bootstrap configs
  --newjson NEWJSON     Provide JSON input for bootstrap variables; mutually
                        exclusive with custom and bootstrap configs
  --log_dir LOG_DIR     Log Directory to for logfiles
  --multi_use           Add options for use in multiple use-cases
  --console_type_serial {true,false}
                        Attach the console serial to the VM; default is false;
                        --console_type_serial=true/false;
  --root_file_disk_bus {virtio,ide}
                        root disk file type: --root_file_disk_bus=virtio/ide;
                        default is virtio
  --virtual_interface_model {rtl8139}
                        --virtual_interface_model=rtl8139; default is none
  --thick_disk_provisioning {true,false}
                        --thick_disk_provisioning=true; default is false
  --eager_zero {true,false}
                        --eager_zero=true; default is false
  --nocloud {true,false}
                        --nocloud=true/false; default is false
  --bootstrap_cloud_init_bus_type {ide,virtio}
                        --bootstrap_cloud_init_bus_type=virtio; default is ide
  --bootstrap_cloud_init_drive_type {cdrom,disk}
                        --bootstrap_cloud_init_drive_type=disk; default is
                        cdrom
```

```
  --bootstrap BOOTSTRAP
                        Every bootstrap file should be a different option Non
                        HA format: --bootstrap
                        <mountpoint>:<file1>,<mountpoint>:<file2>... See
                        usage.txt for more details HA format for SDWAN
                        NetworkHub: --bootstrap mount_point:<value>,file:<file
                        2mount>[,<attrib>:<value>] mount_point:<value> and
                        file:<file2mount> are mandatory followed by one or
                        more attributes in the format <attrib>:<value>
  --interface_hot_add {true,false}
                        VM supports interface add without power off. Default
                        is set to true; --interface_hot_add=true/false
  --interface_hot_delete {true,false}
                        VM supports interface delete without power off.
                        Default is set to false;
                        --interface_hot_delete=true/false
  -v, --verbose         verbose
  -q, --quiet           quiet
  --no_compress         creates tar file without compressing the input files
  --cleanup             deletes all the input and configuration files upon tar
                        file created
  --tablet {true,false}
                        : Add input device of type tablet --tablet=true/false;
  --ha_package          enable HA packaging
  --mgmt_vnic MGMT_VNIC
                        VM management interface identifier
  --pack_dir <DIR> PACK
                        package all files in directory

Required:
  -o PACKAGE_FILENAME, --package_filename PACKAGE_FILENAME
                        [REQUIRED] file name for the target VNF package name-
                        default is root disk image name with extension .tar.gz
  -i ROOT_DISK_IMAGE, --root_disk_image ROOT_DISK_IMAGE
                        [REQUIRED] List of root disk images to be bundled
                        example: --root_disk_image isrv.qcow2;
                        --root_disk_image isrv1.qcow2,isrv2.qcow2
  --prop_template PROP_TEMPLATE
                        image properties template file name including path
                        default path is the current dir of the tool and name
                        is image_properties_template.xml if the user doesn't
                        input this option example: --prop_template
                        /usr/bin/image_properties_template.xml
  -t VNF_TYPE, --vnf_type VNF_TYPE
                        [REQUIRED] VNF type, e.g. ROUTER, FIREWALL, vWAAS,
                        vWLC, and OTHER
  -n NAME, --vnf_name NAME
                        [REQUIRED] Name of the VNF image
  -r VNF_VERSION, --vnf_version VNF_VERSION
                        [REQUIRED] VNF version, e.g. --vnf_version 1.0 or
                        --vnf_version 0.9
  --app_vendor APP_VENDOR
                        Application Vendor e.g. Cisco, Juniper etc
  --monitored {true,false}
                        [REQUIRED] Monitored VNF: --monitored=true/false;
  --optimize {true,false}
                        [REQUIRED] optimized VM: --optimize=true/false;

HA options:
  --ha_capable
  --ha_vnic HA_VNIC     VM HA vnic
  --ha_vnic_count HA_VNIC_COUNT
                        Number of ha_vnics
```

```
Resources:
  Resources: min and max - vCPU, memory and disk

  --min_vcpu VCPU_MIN   min #vCPU : min number of vCPU supported by VM
                        example:--min_vcpu 2
  --max_vcpu VCPU_MAX   max #vCPU : max number if vCPU required for VM
                        example:--max_vcpu 4
  --min_mem MEMORY_MB_MIN
                        min mem : min mem in MB required for VM
                        example:--min_mem 1024
  --max_mem MEMORY_MB_MAX
                        max mem : max mem in MB required for VM
                        example:--max_mem 4196
  --min_disk ROOT_DISK_GB_MIN
                        min disk : min disk in GB required for VM
                        example:--min_disk 8
  --max_disk ROOT_DISK_GB_MAX
                        max disk : max disk in GB required for VM
                        example:--max_disk 8
  --vnic_max VNIC_MAX   max number of Vnics allowed for VM example:--vnic_max
                        8
  --vnic_names VNIC_NAMES
                        list of vnic number to name mapping in format
                        number:name example --vnic_names
                        1:GigabitEthernet2,2:GigabitEthernet4

Profile Options:
  --profile PROFILE     enter the profile name, profile description, no of
                        vCPU required, min memory required in MB, min disk
                        space required in MB, example: --profile
                        profile1,"This is profile 1",2,2048,4096 --profile
                        profile2,"This is profile 2",4,4096,4096
  --default_profile DEFAULT_PROFILE
                        default profile

Driver Support Options:
  --sriov {true,false}  Enable/Disable SRIOV support: --sriov=true/false;
                        default is false
  --sriov_list SRIOV_DRIVER_LIST
                        list of SRIOV drivers example: --sriov_list
                        igb,igbvf,i40evf
  --pcie {true,false}   Not supported
  --pcie_list PCIE_DRIVER_LIST
                        Not supported

Privilege/Priority Options:
  --privileged {true,false}
                        Not supported

Custom Properties:
  --custom CUSTOM       custom properties format: --custom ["propattr_"<attr>:
                        <value>],key:<value>,[keyattr_<attr>:<value>],type:<va
                        lue>,val<N>:<value>,[val<N>attr_<attr>:<value>] Allows
                        specification of custom properties: 0 or more
                        propattr_<attr>:<value> pairs - 'propattr' is a
                        keyword and used to specify property attributes
                        key:<value> pairs 0 or more keyattr_<attr>:value pairs
                        - 'keyattr' is a keyword and is used to specify key
                        attributes type:<value> pair - type of value
                        valN:<value> pair - val1:value,val2:value etc 0 or
                        more valNattr_<attr>:<value> pairs - 'val<N>attr' is
                        an attribute for val<N> See usage_examples.txt
```

The table lists the parameters that can be passed to the **nfvpt.py** command.

Parameter	Mandatory/Optional	Description
version	Not applicable	Show program's version number and exit.
help	Not applicable	Show this help message and exit.
package_file_name	Mandatory	File name for the target VNF package. The default is the root disk image name with extension *.tar.gz*.
disk_img_names	Mandatory	List of root disk images to be bundled. Only the qcow2 images are supported.
img_name	Mandatory	Name of the VNF image.
vnf_type	Mandatory	VNF type Supported types are: ROUTER, FIREWALL, vWAAS, vWLC, and OTHER.
vnf_version	Mandatory	VNF version
monitored	Mandatory	VM health monitoring for those VMs that can be bootstrapped Options are: true/false Monitoring timeout period for a monitored VM is 600 seconds by default
optimize	Mandatory	Optimized VM Options are: true/false
virtual_interface_model	Optional	Default is none.
thick_disk_provisioning	Optional	Default is false.
eager_zero	Optional	Default is false.
bootstrap_cloud_init_bus_type	Optional	Default is IDE.
bootstrap_cloud_init_drive_type	Optional	Mounts the day0 configuration file as disk Default is CD-ROM.
bootstrap	Optional	Bootstrap files for VNF. Two parameters are required in the format of dst:src; dst filename including path has to match exactly to what the VM expects; up to 20 bootstrap files are accepted. For example: --bootstrap ovf-env.xml for ISRv and --bootstrap day0-config for ASAv.

Parameter	Mandatory/Optional	Description
min_vcpu	Optional	Minimum number of vCPUs supported by the VM. The default is 1.
max_vcpu	Optional	Maximum number of vCPUs required for the VM. The default is 8.
min_mem	Optional	Minimum memory in MB required for the VM. The default is 4 GB.
max_mem	Optional	Maximum memory in MB required for the VM. Physical memory: 2 GB The default is 8 GB.
min_disk	Optional	Minimum disk in GB required for the VM. The default is 8 GB.
max_disk	Optional	Maximum disk in GB required for the VM. Available disks are SSD and HDD: 15 GB The default is 16 GB
vnic_max	Optional	Maximum number of VNICs allowed for the VM. The default is 8.
profile	Optional	The profile name, profile description, number of vCPUs required, minimum memory required in MB and minimum disk space required in MB.
default_profile	Optional	The default profile.
sriov	Optional	Enable or disable SRIOV support. The default is false.
sriov_list	Optional	List of SRIOV drivers.
pcie	Optional	Not supported.
pcie_list	Optional	Not supported.
privileged	Optional	Not supported.

Parameter	Mandatory/Optional	Description
custom	Optional	Custom properties to be supported and/or passed to the bootstrap configuration with tokenized variables. This is only used for the local portal to display options for the user to choose while deploying.
pack_dir	Optional	package all files in directory

NFVIS Specific Enhancements

Note Use pack_dir option if the *.tar.gz already exists and you want to modify the bootstrap configuration file or image_properties.xml manually.

The following parameters are added as part of the NFVIS specific enhancements:

```
 --pack_dir <DIR> PACK

                       package all files in directory

Resources:

  --vnic_names VNIC_NAMES

                       list of vnic number to name mapping in format

                       number:name example --vnic_names

                       1:GigabitEthernet2,2:GigabitEthernet4
```

Usage

Follow the steps to change a single line in day-0 configuration file or add a single option in image_properties.xml:

1. Get the working VM packaging image - isrv*.tar.gz.
2. Extract the contents - tar -xvf isrv*.tar.gz.
3. Modify the file contents as required.
4. nfvpt.py --pack_dir current-working-dir-with-files -i isrv.qcow2 -o isrv.tar.gz

VM Packaging Utility Usage Examples

Given below are the contents of the file *nfvis_vm_packaging_utility_examples.txt*:

Example 1: Usage for TinyLinux

```
nfvpt.py -o TinyLinux -i TinyLinux.qcow2 -n TinyLinux -t linux -r 1.0 --monitored false
--min_vcpu 1 --max_vcpu 2 --min_mem 1024 --max_mem 1024 --min_disk 1 --max_disk 2
--vnic_max 1 --optimize false
```

Example 2: Usage for ASAv

Note The bootstrap filename has to be *day0-config*. This cannot be modified as ASAv looks for the exact filename.

```
nfvpt.py -o asav961-201 -i asav961-201.qcow2 -n ASAv -t firewall -r 961-201 --monitored
true --bootstrap day0-config:filename1
--min_vcpu 1 --max_vcpu 4 --min_mem 1024 --max_mem 8192 --min_disk 8 --max_disk 16 --vnic_max
 8 --optimize true
--profile ASAv5,"ASAv5 profile",1,1024,8192 --profile ASAv10,"ASAv10 profile",1,4096,8192
--profile ASAv30,"ASAv30 profile",4,8192,16384
--default_profile ASAv5
```

Example 3: Usage for ISRv

Note The bootstrap filename has to be *ovf-env.xml* . This cannot be modified as ISRv looks for the exact filename.

```
nfvpt.py -o isrv.16.03.01 -i isrv-universalk9.16.03.01.qcow2 -n ISRv.16.03.01 -t ROUTER -r
 16.03.01 --monitored true --privileged true
--bootstrap ovf-env.xml:file1,ios-xe.txt:file2 --min_vcpu 2 --max_vcpu 8 --min_mem 4096
--max_mem 8192 --min_disk 8 --max_disk 8
--vnic_max 8 --optimize true --profile ISRv-small,"ISRv small profile",2,4096,8192 --profile
 ISRv-medium,"ISRv medium profile",4,4096,8192
--default_profile ISRv-small --sriov_list igb,igbvf,i40evf --custom tech_package,ax
```

Example 4: Usage for a third party VM with config drive (ISO) mounted at specific path on the VM:

```
nfvpt.py -o test.1.0 -i test-1.0.qcow2 -n TEST -t OTHER -r 1.0 --monitored true --privileged
 true
--bootstrap /:bootstrap.xml,/license/lic.txt:license.txt --min_vcpu 2 --max_vcpu 8 --min_mem
 4096 --max_mem 8192
--min_disk 8 --max_disk 8 --vnic_max 8 --optimize true --profile small,"small
profile",2,4096,8192
--profile medium,"medium profile",4,4096,8192 --default_profile small
```

In this case, *test.1.0.pkg : bootstrap.xml* gets mounted as *bootstrap.xml* at the root, and the *license.txt* gets mounted as */license/lic.txt*.

Example 5: Usage for Palo Alto Firewall

```
nfvpt.py -o PA_L3_HA -i PA-VM-KVM-8.0.5.qcow2 --json d.json -t firewall -n "PA FIREWALL"
-r 8.0.5 --app_vendor PA --monitor true --ha_package
```

Example 6: Usage for Asav

```
nfvpt.py -i foo.qcow2 -o asav.tar.gz --json pa1.json --app_vendor cisco -t firewall -r 10
--optimize true -n asav --monitored true --ha_package —ha_capable
```

Example 7: Usage for csr

```
nfvpt.py --ha_package --pack_dir /data/intdatastore -i csr1000v-universalk9.16.09.01.qcow2
  -o csr1000v-universalk9.16.09.01-ha.tar.gz
```

Standard VM Image Packaging

The standard VM packaging is based on the Open Virtualization Format (OVF) packaging standard, in which a single file is distributed in open virtualization appliance (OVA) format. The VM image is shared using a TAR archive file with the root disk image and descriptor files.

Note Cisco Enterprise NFVIS supports VM packaging in *.tar.gz* (compressed form of OVA) format. Ensure that all supported third party VM images are available in the supported format.

Generating a VM Package

Package files are provided for Cisco ISRv, Cisco ASAv, and tiny Linux and Windows server 2000. Vendors are responsible for packaging all third party VMs in the supported format.

1. Create a VM qcow2 image.
2. Create an *image_properties.xml* file with the VM properties. Ensure that you add all mandatory fields. Include the profiles supported for the VM in this file, and select one default profile. If you do not want to monitor the VM bootup, make the bootup time as -1.
3. Create *bootstrap-config* or *day0-config*, if any bootstrap configuration is required for the VM. If the bootstrap configuration requires inputs from the user, use the tokens in the xml or text file. These tokens are populated during the VM deployment with the provided data.

   Note A VM deployment may fail, if there are tokens in the configuration, and the user does not provide the token values in the deployment payload.
4. Create a *package.mf* file, which lists all the files to be bundled into the *.tar.gz* file along with checksums.
5. Generate the packaging file using "tar -cvzf ova_file_name list_of_files_to_be_bundled".

   For example, *tar -cvzf isrv.tar.gz isrv-universalk9.03.16.02.S.155-3.S1a-ext-serial.qcow2 image_properties.xml isr_ovf_env.xml package.mf*.

Appendix

VM Image Package Files

The table lists the contents of the VM package that are generated using the packaging tool:

Table 6: VM Image Package Files

File	Description	Mandatory/Optional
Package Manifest (package.mf)	Lists the files in the package and the expected checksum for the files.	Mandatory

VM image properties (vmname_properties.xml)	XML file with resources and features supported by the VM	Mandatory
VM image (vmname.qcow2)	Image file of the VM. Multiple images are supported. One root_disk image file is mandatory.	Mandatory
Bootstrap (bootstrap_file)	Optional	Bootstrap files for VNF. Two parameters are required in the format of dst:src; dst filename including path has to match exactly to what the VM expects; up to 20 bootstrap files are accepted. For example: --bootstrap ovf-env.xml for ISRv and --bootstrap day0-config for ASAv.

Package Manifest File

The package manifest XML file provides a list of the files in the package with their names and their expected checksum. SHA1 algorithm (sha1sum) is used to calculate the checksum. This is a mandatory file to be bundled in the VM package. The manifest file must be named as *package.mf*.

Table 7: Package Manifest File Details

Property Name	Description	Property Tag	Mandatory/Optional
File information	XML tree with details of file name, file type, and expected checksum. The root_image and image_properties files are required.	<file_info>	Mandatory
File name	Name of the file	<name>	Mandatory
File type	Describes the file type. Supported types: • root_image • image_properties • bootstrap_config_file • ephemeral_disk1_image • ephemeral_disk2_image	<type>	Mandatory
Expected checksum	The calculated SHA1 checksum to be validated.	<sha1_checksum>	Mandatory

Bootstrap Configuration File

The bootstrap configuration file is an XML or a text file, and contains properties specific to a VM and the environment. Properties can have tokens, which can be populated during deployment time from the deployment payload.

VM Image Properties File

This XML file provides information about the resources supported or required for the VM operation. All mandatory parameters have to be defined. It also supports custom attributes. This is a mandatory file to be bundled in the VM package. The VM package supports up to 10 disks to be bundled into the package.

Table 8: VM Image Properties File Details

Property Name	Description	Property Tag	Possible Values	Mandatory/Optional
VNF Type	VM functionality provided. Router and firewall are predefined types.	<vnf_type>	Router, firewall, Windows, Linux, and custom_type	Mandatory
Name	Name associated with the VM packaging. This name is referenced for VM deployment.	<name>	Any	Mandatory
Version	Version of the package	<version>	Any	Mandatory
Boot-up time	Boot-up time (in seconds) of the VNF before it can be reachable via ping.	<bootup_time>	Any in seconds, (-1) to not monitor boot-up	Mandatory
Root Disk Image Bus	Root image disk bus	<root_file_disk_bus>	virtio, scsi, and ide	Mandatory
Disk-1 bus type	Additional disk1 image disk bus	<disk_1_file_disk_bus>	virtio, scsi, and ide	Optional
Disk-2 bus type	Disk2 image disk bus	<disk_2_file_disk_bus>	virtio, scsi, and ide	Optional
Disk-10 bus type	Disk10 image disk bus	<disk_10_file_disk_bus>	virtio, scsi, and ide	Optional
Root Disk Image format	Root image disk format	<root_image_disk_format>	qcow2 and raw	Mandatory
Disk-1 Image format	Additional disk 1 image format	<disk_1_image_format>	qcow2 and raw	Optional

Disk-2 Image format	Disk 2 image format	<disk_2_image_format>	qcow2 and raw	Optional
Disk-10 Image format	Disk 10 image format	<disk_10_image_format>	qcow2 and raw	Optional
Serial Console	Serial console supported	<console_type_serial>	true, false	Optional
Minimum vCPU	Minimum vCPUs required for a VM operation	<vcpu_min>		Mandatory
Maximum vCPU	Maximum vCPUs supported by a VM	<vcpu_max>		Mandatory
Minimum memory	Minimum memory in MB required for VM operation	<memory_mb_min>		Mandatory
Maximum memory	Maximum memory in MB supported by a VM	<memory_mb_max>		Mandatory
Minimum root disk size	Minimum disk size in GB required for VM operation	<root_disk_gb_min>		Optional
Maximum root disk size	Maximum disk size in GB supported by a VM	<root_disk_gb_max>		Optional
Maximum vNICs	Maximum number of vNICs supported by a VM	<vnic_max>		Mandatory
SRIOV support	SRIOV supported by VM interfaces. This should have a list of supported NIC device drivers.	<sriov_supported>	true, false	Optional
SRIOV driver list	List of drivers to enable SRIOV support	< sriov_driver_list>		Optional

PCI passthru support	PCI passthru support by VM interfaces	<pcie_supported>	true, false	Optional
PCIE driver list	List of VNICS to enable PCI passthru support	< pcie _driver_list>		Optional
bootstrap_cloud_init_drive_type	Mounts day0 config file as disk (default is CD-ROM)	<bootstrap_cloud_init_drive_type>	disk, cdrom	Optional
bootstrap_cloud_init_bus_type	Default is IDE	<bootstrap_cloud_init_bus_type>	virtio, ide	Optional
BOOTSTRAP	Bootstrap files for the VNF. Two parameters are required in the format of dst:src; dst filename including path has to match exactly to what the VM expects; up to 20 bootstrap files are accepted. For example: --bootstrap ovf-env.xml for ISRv and --bootstrap day0-config for ASAv	< bootstrap_file>	File name of the bootstrap file	Optional

Custom properties	List of properties can be defined within the custom_property tree. (Example: For ISRv, the technology packages are listed in this block.) If the Cisco Enterprise NFV portal is used to deploy the VM, the portal prompts you for inputs for custom properties fields, and can pass the values to the bootstrap configuration.	<custom_property>		Optional
Profiles for VM deployment	List of VM deployment profiles. Minimum one profile is required	<profiles>		Optional
Default profile	The default profile is used when no profile is specified during deployment.	<default_profile>		Optional
Monitoring Support	A VM supports monitoring to detect failures.	<monitoring_supported>	true, false	Mandatory
Monitoring Method	A method to monitor a VM. Currently, only ICMP ping is supported.	<monitoring_methods>	ICMPPing	Mandatory if monitoring is true
Low latency	If a VM's low latency (for example, router and firewall) gets dedicated resource (CPU) allocation. Otherwise, shared resources are used.	<low_latency>	true, false	Mandatory

Privileged-VM	Allows special features like promiscuous mode and snooping . By default, it is false.	<privileged_vm>	true, false	Optional
Virtual interface model		<virtual_interface_model>		Optional
Thick disk provisioning	By default, it is false.	<thick_disk_provisioning>	true, false	Optional
Profile for VM deployment	A profile defines the resources required for VM deployment. This profile is referenced during VM deployment.	<profile>		Optional
Name	Profile name	<name>	Any	Mandatory
Description	Description of the profile	<description>	Any	Mandatory
vCPU	vCPU number in a profile	<vcpus>		Mandatory
Memory	Memory - MB in profile	<memory_mb>		Mandatory
Root Disk Size	Disk size - MB in profile .	<root_disk_mb>		Mandatory
VNIC Offload	List of properties that can be set for vnic offload	<vnic_offload>		Optional
Generic Segmentation Offload	Turn generic segmentation offload on or off	<generic_segmentation_offload> (parent: <vnic_offload>)	on, off	Optional
Generic Receive Offload	Turn generic receive offload on or off	<generic_receive_offload> (parent: <vnic_offload>)	on, off	Optional
RX Checksumming	Turn RX checksumming on or off	<rx_checksumming> (parent: <vnic_offload>)	on, off	Optional
TX Checksumming	Turn TX checksumming on or off	<tx_checksumming> (parent: <vnic_offload>)	on, off	Optional

TCP Segmentation Offload	Turn TCP segmentation offload on or off	<tcp_segmentation_offload> (parent: <vnic_offload>)	on, off	Optional

Note A virtual console is supported by default. Specify the root disk size as zero for multiple disks (for example, vWaas deployment) as the system does not support populating multiple disk sizes. Actual disk sizes are calculated from the root_disk files.

Example: Package.mf

```
** sha1sum - for calculating checksum
<PackageContents>
  <File_Info>
    <name>ISRv_serial_3.16.02.qcow2</name>
    <type>root_image</type>
    <sha1_checksum>93de73ee3531f74fddf99377972357a8a0eac7b</sha1_checksum>
  </File_Info>
  <File_Info>
    <name>image_properties.xml</name>
    <type>image_properties</type>
    <sha1_checksum>c5bb6a9c5e8455b8698f49a489af3082c1d9e0a9</sha1_checksum>
</File_Info>
  <File_Info>
    <name>ISRv_ovf_env.xml</name>
    <type> bootstrap_file_1</type>
    <sha1_checksum>c5bb6a9c5e8455b8698f49a489af3082c1d9e0a9</sha1_checksum>
   </File_Info>
   <File_Info>
     <name>ISRv_disk1_image.qcow2</name>
     <type>ephemeral_disk1_image</type>
     <sha1_checksum>aac24513098ec6c2f0be5d595cd585f6a3bd9868</sha1_checksum>
   </File_Info>
</PackageContents>
```

Example: Image Properties

```
<?xml version="1.0" encoding="UTF-8"?>
<image_properties>
    <vnf_type>ROUTER</vnf_type>
    <name>isrv-universalk9</name>
    <version>03.16.02</version>
    <bootup_time>600</ bootup_time >
    <root_file_disk_bus>virtio</root_file_disk_bus>
    <root_image_disk_format>qcow2</root_image_disk_format>
    <vcpu_min>1</vcpu_min>
    <vcpu_max>8</vcpu_max>
    <memory_mb_min>4096</memory_mb_min>
    <memory_mb_max>8192</memory_mb_max>
    <vnic_max>8</vnic_max>
    <root_disk_gb_min>8</root_disk_gb_min>
    <root_disk_gb_max>8</root_disk_gb_max>
    <console_type_serial>true</console_type_serial>
    <sriov_supported>true</sriov_supported>
    <sriov_driver_list>igb</sriov_driver_list>
    <sriov_driver_list>igbvf</sriov_driver_list>
```

```
    <sriov_driver_list>i40evf</sriov_driver_list>
    <pcie_supported>true</pcie_supported>
    <pcie _driver_list> igb </pcie_driver_list>
    <pcie _driver_list> igbvf</pcie_driver_list>
    <pcie _driver_list> i40evf</pcie_driver_list>
    <bootstrap_file_1> ovf-env.xml </bootstrap_file_1>
    <monitoring_supported>true</monitoring_supported>
    <monitoring_methods>ICMPPing</monitoring_methods>
    <low_latency>true</low_latency>
    <privileged_vm>true</privileged_vm>
    <cdrom>true</cdrom>
    <custom_property>
        <tech_package>ax</tech_package>
        <tech_package>sec</tech_package>
        <tech_package>ipbase</tech_package>
        <tech_package>appx</tech_package>
    </custom_property>
    <profiles>
         <profile>
            <name>ISRv1kv-small</name>
            <description>ISRv upto 50MBPS performance</description>
            <vcpus>1</vcpus>
            <memory_mb>4096</memory_mb>
            <root_disk_mb>8</root_disk_mb>
         </profile>
         <profile>
            <name>ISRv1kv-medium</name>
            <description>ISRv upto 250MBPS performance</description>
            <vcpus>2</vcpus>
            <memory_mb>4096</memory_mb>
            <root_disk_mb>8</root_disk_mb>
         </profile>
    </profiles>
    <default_profile>small</default_profile>
 </image_properties>
```

Example: Bootstrap Configuration File

```
<?xml version="1.0" encoding="UTF-8"?>
<Environment
xmlns:oe="http://schemas.dmtf.org/ovf/environment/1">
  <PropertySection>
    <Property oe:key="com.cisco.ISRv.config-version.1" oe:value="1.0"/>
    <Property oe:key="com.cisco.isrv.enable-ssh-server.1" oe:value="True"/>
    <Property oe:key="com.cisco.isrv.login-password.1" oe:value="admin"/>
    <Property oe:key="com.cisco.isrv.login-username.1" oe:value="lab"/>
    <Property oe:key="com.cisco.isrv.mgmt-interface.1" oe:value="GigabitEthernet1"/>
   <Property oe:key="com.cisco.isrv.mgmt-ipv4-addr.1" oe:value="${NICID_0_IP_ADDRESS}/24"/>

    <Property oe:key="com.cisco.isrv.mgmt-ipv4-network.1" oe:value=""/>
    <Property oe:key="com.cisco.isrv.license.1" oe:value="${TECH_PACKAGE}"/>
   <Property oe:key="com.cisco.isrv.ios-config-0001" oe:value="vrf definition Mgmt-intf"/>

    <Property oe:key="com.cisco.isrv.ios-config-0002" oe:value="address-family ipv4"/>
    <Property oe:key="com.cisco.isrv.ios-config-0003" oe:value="exit-address-family"/>
    <Property oe:key="com.cisco.isrv.ios-config-0004" oe:value="address-family ipv6"/>
    <Property oe:key="com.cisco.isrv.ios-config-0005" oe:value="exit-address-family"/>
    <Property oe:key="com.cisco.isrv.ios-config-0006" oe:value="exit"/>
   <Property oe:key="com.cisco.isrv.ios-config-0007" oe:value="interface GigabitEthernet1"/>

    <Property oe:key="com.cisco.isrv.ios-config-0008" oe:value="vrf forwarding Mgmt-intf"/>
```

```
    <Property oe:key="com.cisco.isrv.ios-config-0009" oe:value="ip address
${NICID_0_IP_ADDRESS} ${NICID_0_NETMASK}"/>
    <Property oe:key="com.cisco.isrv.ios-config-0010" oe:value="no shut"/>
    <Property oe:key="com.cisco.isrv.ios-config-0011" oe:value="exit"/>
    <Property oe:key="com.cisco.isrv.ios-config-0012" oe:value="ip route vrf Mgmt-intf
0.0.0.0 0.0.0.0 ${NICID_0_GATEWAY}"/>
  </PropertySection>
</Environment>
```

Image Properties Template File

The parameters that go into the image properties file are listed in the code extract below.

```
<?xml version="1.0" encoding="UTF-8"?>
<image_properties>
    <vnf_type>ROUTER</vnf_type>
    <name>TEMPLATE</name>
    <version>1.0</version>
    <bootup_time>600</bootup_time>
    <root_file_disk_bus>virtio</root_file_disk_bus>
    <root_image_disk_format>qcow2</root_image_disk_format>
    <vcpu_min>1</vcpu_min>
    <vcpu_max>8</vcpu_max>
    <memory_mb_min>4096</memory_mb_min>
    <memory_mb_max>8192</memory_mb_max>
    <vnic_max>8</vnic_max>
    <root_disk_gb_min>8</root_disk_gb_min>
    <root_disk_gb_max>16</root_disk_gb_max>
    <console_type_serial>false</console_type_serial>
    <sriov_supported>true</sriov_supported>
    <sriov_driver_list>s1</sriov_driver_list>
    <sriov_driver_list>s2</sriov_driver_list>
    <sriov_driver_list>s3</sriov_driver_list>
    <pcie_supported>false</pcie_supported>
    <monitoring_supported>true</monitoring_supported>
    <monitoring_methods>ICMPPing</monitoring_methods>
    <low_latency>true</low_latency>
    <privileged_vm>false</privileged_vm>
    <cdrom>true</cdrom>
    <bootstrap_file_1>b1.xml</bootstrap_file_1>
    <bootstrap_file_2>b2.txt</bootstrap_file_2>
    <custom_property>
        <key>val</key>
    </custom_property>
    <profiles>
        <profile>
            <name>small</name>
            <description>small</description>
            <vcpus>1</vcpus>
            <memory_mb>1024</memory_mb>
            <root_disk_mb>4096</root_disk_mb>
        </profile>
        <profile>
            <name>medium</name>
            <description>medium</description>
            <vcpus>2</vcpus>
            <memory_mb>4096</memory_mb>
            <root_disk_mb>8192</root_disk_mb>
        </profile>
```

```
    </profiles>
    <default_profile>small</default_profile>
</image_properties>
```

Image Registration

To register a VM image, you must first copy or download the relevant VM image to the NFVIS server, or host the image on a http or https server. Once you have downloaded the file, you can register the image using the registration API. The registration API allows you to specify the file path to the location (on the http/https server) where the tar.gz file is hosted. Registering the image is a one-time activity. Once an image is registered on the http or https server, and is in active state, you can perform multiple VM deployments using the registered image. All VM images are available in VM packaging and VM package content. For more information see, VM Image Packaging Utility, on page 68

Register VM Packages Using REST API

This example shows the sequence of registering a tar.gz package on Cisco Enterprise NFVIS using REST API.

Post Image Registration

```
curl -v -u -k 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/vm_lifecycle/images -d
'<image><name>WinServer2012R2.iso</name><src>file:///data/intdatastore/uploads/WinServer2012R2.iso/WinServer2012R2.iso</src></image>'

HTTP/1.1 201 Created
```

Get Image Status

```
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/vm_lifecycle/opdata/images/image/isrv-03.16.02?deep

HTTP/1.1 200 OK
<image xmlns="http://www.cisco.com/nfvis/vm_lifecycle" xmlns:y="http://tail-f.com/ns/rest"
xmlns:esc="http://www.cisco.com/nfvis/vm_lifecycle">
<name>isrv.03.16.02</name>
<image_id>585a1792-145c-4946-9929-e040d3002a59</image_id>
<public>true</public>
<state>IMAGE_ACTIVE_STATE</state></image>
```

Get Registered Image Status

```
Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X
GET https://209.165.201.1/api/config/vm_lifecycle/images?deep

HTTP/1.1 200 OK
<images xmlns="[http://www.cisco.com/esc/esc|http://www.cisco.com/nfvis/vm_lifecycle]"
xmlns:y="[http://tail-f.com/ns/rest|http://tail-f.com/ns/rest]" 
xmlns:esc="[http://www.cisco.com/nfvis/vm_lifecycle|http://www.cisco.com/nfvis/vm_lifecycle]">
<image>
<name>isrv-9.16.03.01</name>
<src>http://data/nfvos-pkg/isr/isrv-universalk9.16.03.01.tar.gz</src>
```

```
</image>
</images>
```

Delete Registered Image

```
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/vm_lifecycle/images/image/isrv-3.16.0.1a

HTTP/1.1 204 No Content
```

For more information on REST APIs related to image registration, see API Reference for Cisco Enterprise NFVIS.

Register VM Image with Multiple Root Disks

If any image requires multiple root disks, you can specify it in the image properties file.

This example shows how to specify multiple root disks in image properties.

```
<image_properties>
 …
 <root_file_disk_bus>virtio</root_file_disk_bus>
 <root_image_disk_format>qcow2</root_image_disk_format>
 <disk_1_file_disk_bus>virtio</ disk_1_file_disk_bus>
 <disk_1_image_format>qcow2</ disk_1_image_format>
 <disk_2_file_disk_bus>virtio</ disk_2_file_disk_bus>
 <disk_2_image_format>qcow2</ disk_2_image_format>
 …
</image_properties>
```

Register a VM Image through a Root Disk

A VM can also be registered using just a disk image (qcow2 or iso) without packaging into a tar.gz. As there will be no image properties in this scenario, the default image properties are used.

Default Properties for Root Disk Registration

The following table lists the default properties that are provisioned if you register a VM image by uploading the root disk for the image.

Property Name	Property Tag	Default Value
Version	<version>	NA
VNF Type	<vnf_typ>	OTHER
VCPU Min	<vcpu_min>	1
VCPU Max	<vcpu_max>	64
Memory Min (MB)	<memory_mb_min>	256
Memory Max (MB)	<memory_mb_max>	1048576
Root Disk Min Size (GB)	<root_disk_gb_min>	1

Property Name	Property Tag	Default Value
Root Disk Max Size (GB)	<root_disk_gb_max>	10240
VNIC Max	<vnic_max>	8
Bootup Time	<bootup_time>	-1
Interface Hot Add	<interface_hot_add>	true
Interface Hot Delete	<interface_hot_delete>	false

Register a Remote VM Image

Cisco Enterprise NFVIS allows you to register a VM image that is stored at a remote location or a web server. by specifying the URL to the image location in the source field.

If the web server supports HTTPS, the you can choose to enable Certificate Validation to validate its authenticity. Certificate Validation requires the server's public key to be specified, either in string or file format, in the image registration payload. This allows NFVIS to perform asymmetric encryption and download/register the image file securely over HTTPS.

Example: POST Remote Image Registration from Webserver over HTTPS

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H
Accept:application/vnd.yang.data+xml -H Content-
Type:application/vnd.yang.data+xml -X POST
https://172.29.91.28/api/config/vm_lifecycle/images/ -d '
<image>
 <name>ASAV</name>
 <src>https://172.20.117.124/nfvis/asav982.tar.gz</src>
</image>'

HTTP/1.1 201 Created
```

Example: POST Remote Image Registration from Webserver over HTTPS

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H
Accept:application/vnd.yang.data+xml -H Content-
Type:application/vnd.yang.data+xml -X POST
https://172.29.91.28/api/config/vm_lifecycle/images/ -d '
<image>
 <name>ASAV</name>
 <src>https://172.20.117.124/nfvis/asav982.tar.gz</src>
 <certificate_validation>true</certificate_validation>
 <certificate_file>/data/intdatastore/uploads/pub_key.cert</certificate_file>
</image>'

HTTP/1.1 201 Created
```

Specify Storage Location for a VM Image

Cisco Enterprise NFVIS allows users to specify the location where the register image should be stored, using the *placement* property tag.

The following table lists the placement values supported and their respective mappings.

storage name	directory map
datastore1	/data
datastore2	/mnt/extdatastore1
datastore3	/mnt/extdatastore2
nfs:nfs_mount_name	/data/mount/nfs/nfs_mount_name
nfs or nfs:nfs	/data/mount/nfs/
nfs:nfs_storage or nfs_storage	/data/mount/nfs_storage

Note If your preferred storage location is **nfs**, you must have it configured to be mounted on NFVIS using appropriate CLIs before registering the image on it.

Example: VM Image Storage Placement

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H
Accept:application/vnd.yang.data+xml -H Content-
Type:application/vnd.yang.data+xml -X POST
https://172.29.91.28/api/config/vm_lifecycle/images/ -d '
<image>
 <name>ASAV</name>
 <src>https://172.20.117.124/nfvis/asav982.tar.gz</src>
 <properties>
  <property>
   <name>placement</name>
   <value>nfs:my_nfs_mount</value>
  </property>
 </properties>
</image>'

HTTP/1.1 201 Created
```

Update VM Image

You can only update the following image properties after a VM image has been registered.

- interface_hot_add
- interface_hot_delete

Any requests to modify other image properties are rejected.

Note When using the REST API, the previously set value of the property must be deleted before updating it with the new value.

Example: Delete Value and Add New Value

1. Delete the previously set property value as shown below.

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H
Accept:application/vnd.yang.collection+xml -X DELETE
https://172.29.91.28/api/config/vm_lifecycle/images/image/ISR_IMAGE/properties/property/interface_hot_add/value

HTTP/1.1 204 No Content
```

2. Add (PUT) the new property value to replace the one you deleted in the previous step.

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H
Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X
PUT
https://172.29.91.28/api/config/vm_lifecycle/images/image/ISR_IMAGE/properties/property/interface_hot_add
 --data '<value>true</value>'

HTTP/1.1 201 Created
```

Example: Get All Properties and Values

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H
Accept:application/vnd.yang.data+xml -H Content-
Type:application/vnd.yang.data+xml -X GET
https://172.29.91.28/api/config/vm_lifecycle/images/image/ISR_IMAGE/properties?deep

HTTP/1.1 200 OK
<properties xmlns="http://www.cisco.com/nfvis/vm_lifecycle"
xmlns:y="http://tail-f.com/ns/rest"  xmlns:vmlc="http://www.cisco.com/nfvis/vm_lifecycle">
  <property>
    <name>interface_hot_add</name>
    <value>true</value>
  </property>
  <property>
    <name>interface_hot_delete</name>
    <value>false</value>
  </property>
</properties>
```

Image Properties

The image_properties.xml file in the tar.gz package contains the property configuration data for a particular image. Figure 3 depicts an example image_properties.xml file.

Some of the properties are mandatory and must be specified to register an image. If any of the mandatory properties are omitted, the image registration fails.

Optional properties can be specified on an image-by-image basis and are not required.

The following table lists all the image properties that are supported in Cisco Enterprise NFVIS.

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
VNF Type	VM functionality provided. Router and firewall are predefined types.	<vnf_type>	Router, firewall, Windows, Linux, and custom_type	Mandatory
Name	Name associated with the VM packaging. This name is referenced for VM deployment.	<name>	Any	Mandatory
Version	Version of the package	<version>	Any	Mandatory
Boot-up time	Boot-up time (in seconds) of the VNF before it can be reachable via ping.	<bootup_time>	Any in seconds, (-1) to not monitor boot-up	Optional, default -1
Root Disk Image Bus	Root image disk bus	<root_file_disk_bus>	virtio, scsi, and ide	Mandatory
Boot Mode	Specifies the mode in which the VNF will boot. Used for Secure Boot feature	<boot_mode>	efi-secure-boot, bios	Optional, default bios
Shim Signature	If using efi-secure-boot boot mode, a shim signature must be provided	<shim_signature>	microsoft, N/A	Required if Boot Mode is specified
Disk-1 bus type	Additional disk1 image disk bus	<disk_1_file_disk_bus>	virtio, scsi, and ide	Optional
Disk-2 bus type	Disk2 image disk bus	<disk_2_file_disk_bus>	virtio, scsi, and ide	Optional
Disk-10 bus type	Disk10 image disk bus	<disk_10_file_disk_bus>	virtio, scsi, and ide	Optional

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
Root Disk Image format	Root image disk format	<root_image_disk_format>	qcow2 and raw	Mandatory
Disk-1 Image format	Additional disk 1 image format	<disk_1_image_format>	qcow2 and raw	Optional
Disk-2 Image format	Disk 2 image format	<disk_2_image_format>	qcow2 and raw	Optional
Disk-10 Image format	Disk 10 image format	<disk_10_image_format>	qcow2 and raw	Optional
Serial Console	Serial console supported	<console_type_serial>	true, false	Optional
Minimum vCPU	Minimum vCPUs required for a VM operation	<vcpu_min>		Mandatory
Maximum vCPU	Maximum vCPUs supported by a VM	<vcpu_max>		Mandatory
Minimum memory	Minimum memory in MB required for VM operation	<memory_mb_min>		Mandatory
Maximum memory	Maximum memory in MB supported by a VM	<memory_mb_max>		Mandatory
Minimum root disk size	Minimum disk size in GB required for VM operation	<root_disk_gb_min>		Optional
Maximum root disk size	Maximum disk size in GB supported by a VM	<root_disk_gb_max>		Optional
Maximum vNICs	Maximum number of vNICs supported by a VM	<vnic_max>		Mandatory

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
SRIOV support	SRIOV supported by VM interfaces. This should have a list of supported NIC device drivers.	<sriov_supported>	true, false	Optional
SRIOV driver list	List of drivers to enable SRIOV support	< sriov_driver_list>		Optional
PCI passthru support	PCI passthru support by VM interfaces	<pcie_supported>	true, false	Optional
PCIE driver list	List of VNICS to enable PCI passthru support	< pcie _driver_list>		Optional
bootstrap_cloud_init_drive_type	Mounts day0 config file as disk (default is CD-ROM)	<bootstrap_cloud_init_drive_type>	disk, cdrom	Optional
bootstrap_cloud_init_bus_type	Default is IDE	<bootstrap_cloud_init_bus_type>	virtio, ide	Optional
BOOTSTRAP	Bootstrap files for the VNF. Two parameters are required in the format of dst:src; dst filename including path has to match exactly to what the VM expects; up to 20 bootstrap files are accepted. For example: --bootstrap ovf-env.xml for ISRv and --bootstrap day0-config for ASAv	< bootstrap_file>	File name of the bootstrap file	Optional

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
Custom properties	List of properties can be defined within the custom_property tree. (Example: For ISRv, the technology packages are listed in this block.) If the Cisco Enterprise NFV portal is used to deploy the VM, the portal prompts you for inputs for custom properties fields and can pass the values to the bootstrap configuration.	<custom_property>		Optional
Profiles for VM deployment	List of VM deployment profiles. Minimum one profile is required	<profiles>		Optional
Default profile	The default profile is used when no profile is specified during deployment.	<default_profile>		Optional
Monitoring Support	A VM supports monitoring to detect failures.	<monitoring_supported>	true, false	Mandatory
Monitoring Method	A method to monitor a VM. Currently, only ICMP ping is supported.	<monitoring_methods>	ICMPPing	Mandatory if monitoring is true

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
Low latency	If a VM's low latency (for example, router and firewall) gets dedicated resource (CPU) allocation. Otherwise, shared resources are used.	<low_latency>	true, false	Mandatory
Privileged-VM	Allows special features like promiscuous mode and snooping . By default, it is false.	<privileged_vm>	true, false	Optional
Disable Spoof Check	Used to disable spoof check for Privledged VMs	<disable_spoof_check>	true, false	Optional
Virtual interface model		<virtual_interface_model>		Optional
Interface Hot Add	If true, an active VNF’s virtual interface can be added/updated without shutting down the VNF.	<interface_hot_add>	true, false	Optional, default true
Interface Hot Delete	If true, an active VNF’s virtual interface can be deleted without shutting down the VNF.	<interface_hot_delete>	true, false	Optional, default false
Thick disk provisioning	During deployment, VM will be a fully allocated root disk with size specified by flavor.	<thick_disk_provisioning>	true, false	Optional, default false

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
Eager Zero	Used in conjunction with Thick disk provisioning. During deployment, root disk is zeroed out to improve I/O operations	<eager_zero>	true, false	Optional, only valid if Thick disk provisioning is enabled. Default false
Profile for VM deployment	A profile defines the resources required for VM deployment. This profile is referenced during VM deployment.	<profile>		Optional
Name	Profile name	<name>	Any	Mandatory
Description	Description of the profile	<description>	Any	Mandatory
vCPU	vCPU number in a profile	<vcpus>		Mandatory
Memory	Memory - MB in profile	<memory_mb>		Mandatory
Root Disk Size	Disk size - MB in profile .	<root_disk_mb>		Mandatory
VNIC Offload	List of properties that can be set for vnic offload	<vnic_offload>		Optional
Generic Segmentation Offload	Turn generic segmentation offload on or off	<generic_segmentation_offload> (parent: <vnic_offload>)	on, off	Optional
Generic Receive Offload	Turn generic receive offload on or off	<generic_receive_offload> (parent: <vnic_offload>)	on, off	Optional
RX Checksumming	Turn RX checksumming on or off	<rx_checksumming> (parent: <vnic_offload>)	on, off	Optional

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
TX Checksumming	Turn TX checksumming on or off	<tx_checksumming> (parent: <vnic_offload>)	on, off	Optional
TCP Segmentation Offload	Turn TCP segmentation offload on or off	<tcp_segmentation_offload> (parent: <vnic_offload>)	on, off	Optional

Example: Contents of an image_properties.xml File

```
<?xml version="1.0" encoding="UTF-8"?>
<image_properties>
    <vnf_type>ROUTER</vnf_type>
    <name>ISRV</name>
    <version>16.06.05</version>
    <bootup_time>600</bootup_time>
    <root_file_disk_bus>virtio</root_file_disk_bus>
    <root_image_disk_format>qcow2</root_image_disk_format>
    <vcpu_min>1</vcpu_min>
    <vcpu_max>8</vcpu_max>
    <memory_mb_min>4096</memory_mb_min>
    <memory_mb_max>8192</memory_mb_max>
    <vnic_max>8</vnic_max>
    <vnic_names>vnics:1:GigabitEthernet2</vnic_names>
    <vnic_names>vnics:2:GigabitEthernet3</vnic_names>
    <vnic_names>vnics:3:GigabitEthernet4</vnic_names>
    <vnic_names>vnics:4:GigabitEthernet5</vnic_names>
    <vnic_names>vnics:5:GigabitEthernet6</vnic_names>
    <vnic_names>vnics:6:GigabitEthernet7</vnic_names>
    <vnic_names>vnics:7:GigabitEthernet8</vnic_names>
    <root_disk_gb_min>8</root_disk_gb_min>
    <root_disk_gb_max>8</root_disk_gb_max>
    <console_type_serial>true</console_type_serial>
    <sriov_supported>true</sriov_supported>
    <sriov_driver_list>igb</sriov_driver_list>
    <sriov_driver_list>igbvf</sriov_driver_list>
    <sriov_driver_list>i40evf</sriov_driver_list>
    <pcie_supported>true</pcie_supported>
    <pcie_driver_list>igb</pcie_driver_list>
    <pcie_driver_list>igbvf</pcie_driver_list>
    <pcie_driver_list>i40evf</pcie_driver_list>
    <monitoring_supported>true</monitoring_supported>
    <monitoring_methods>ICMPPing</monitoring_methods>
    <low_latency>true</low_latency>
    <privileged_vm>true</privileged_vm>
    <cdrom>true</cdrom>
    <bootstrap_file_1>ovf-env.xml</bootstrap_file_1>
    <bootstrap_file_2>iosxe_config.txt</bootstrap_file_2>
    <custom_property>
        <tech_package>ax</tech_package>
        <tech_package>security</tech_package>
        <tech_package>ipbase</tech_package>
        <tech_package>appx</tech_package>
    </custom_property>
    <custom_property>
        <ngio>enable</ngio>
    </custom_property>
    <custom_property>
```

```
        <SSH_USERNAME> </SSH_USERNAME>
    </custom_property>
    <custom_property>
        <SSH_PASSWORD> </SSH_PASSWORD>
    </custom_property>
    <profiles>
     <profile>
     <name>ISRv-mini</name>
            <description>ISRv-mini</description>
     <vcpus>1</vcpus>
            <memory_mb>4096</memory_mb>
            <root_disk_mb>8192</root_disk_mb>
 </profile>
     <profile>
     <name>ISRv-small</name>
            <description>ISRv-small</description>
     <vcpus>2</vcpus>
            <memory_mb>4096</memory_mb>
            <root_disk_mb>8192</root_disk_mb>
 </profile>
     <profile>
     <name>ISRv-medium</name>
            <description>ISRv-medium</description>
     <vcpus>4</vcpus>
            <memory_mb>4096</memory_mb>
            <root_disk_mb>8192</root_disk_mb>
 </profile>
    </profiles>
    <default_profile>ISRv-small</default_profile>
</image_properties>
```

VM Profiles or Flavors

Flavors or profiles define VMs in terms of number of parameters for how to run the VM. Some of the parameters that you can define in a VM profile or flavor are: number of vCPUs, RAM , disk size and so on.

Flavors are created as part of image registration if you use the tar.gz image packages for registering a VM. However, for other image packages such as .qcow2, iso, and raw, you must define custom flavors based on your requirements.

Note Unless specified otherwise in the deployment payload, the value assigned to the custom image property *default_profile* is used at the time of deploying the VM. Only applicable to *tar.gz* image packages.

Example: Create VM Profile Using Rest API

The following example shows how to create a VM profile or flavor using REST API.

```
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/vm_lifecycle/flavors -d '<flavor>
    <name>windows</name>
    <ephemeral_disk_mb>0</ephemeral_disk_mb>
    <memory_mb>4096</memory_mb>
    <root_disk_mb>12288</root_disk_mb>
    <swap_disk_mb>0</swap_disk_mb>
```

```
    <vcpus>2</vcpus>
</flavor>'
```

For more information on REST APIs related to creating VM flavors, see API Reference for Cisco Enterprise NFVIS.

Configure Internal Management Network

Monitored VNFs must have one internal management network specified to the NIC ID “0” in the deployment payload. If the IP address is configured, it needs to belong to the same network as the internal subnet. By default, the internal management network has the range *10.20.0.0*. If the IP address is not specified in the VM deployment payload, the system assigns an IP address for internal management network using IP address pool.

Here an example of configuring internal management network.

```
<interface>
   <nicid>0</nicid>
   <network>int-mgmt-net</network>
   <ip_address>10.20.0.21</ip_address>
</interface>
```

VM Deployment and Management

VM Deployment

A VM can be deployed through API or CLI. Ensure that you have registered a VM image before attempting to deploy it. For more details, see Image Registration, on page 85.

Note The VM name must meet the following requirements:

- Must contain an uppercase character and a lowercase character.
- Must contain a digit.
- Must contain one of the following special characters: dot (.), underscore (_) and hyphen (-).
- Must not have more than 256 characters.

The deployment API allows you to provide values to the parameters that are passed to the system during deployment. Depending on the VM you are deploying, some parameters are mandatory and others optional.

Example: Deploy VMs Using REST API

Use the following API to deploy a VM.

Method	URL
POST	/api/config/vm_lifecycle/tenants/tenant/admin/deployment

Here's a sample payload of deploying a VM.

```
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/vm_lifecycle/tenants /tenant/admin/deployments --data '
<deployment>
  <name>ISRdep</name>
  <vm_group>
    <name>ISRvmgrp</name>
    <image>ISR_IMAGE</image>
    <bootup_time>500</bootup_time>
    <recovery_wait_time>0</recovery_wait_time>
    <interfaces>
      <interface>
        <nicid>0</nicid>
        <network>int-mgmt-net</network>
        <ip_address>10.20.0.21</ip_address>
        <port_forwarding>
          <port>
            <type>ssh</type>
            <protocol>tcp</protocol>
            <vnf_port>22</vnf_port>
            <external_port_range>
              <start>20022</start>
              <end>20022</end>
            </external_port_range>
          </port>
        </port_forwarding>
      </interface>
    </interfaces>
    <kpi_data>
      <kpi>
        <event_name>VM_ALIVE</event_name>
        <metric_value>1</metric_value>
        <metric_cond>GT</metric_cond>
        <metric_type>UINT32</metric_type>
        <metric_collector>
          <type>ICMPPing</type>
          <nicid>0</nicid>
          <poll_frequency>3</poll_frequency>
          <polling_unit>seconds</polling_unit>
          <continuous_alarm>false</continuous_alarm>
        </metric_collector>
      </kpi>
    </kpi_data>
    <rules>
      <admin_rules>
        <rule>
          <event_name>VM_ALIVE</event_name>
          <action>ALWAYS log</action>
          <action>TRUE servicebooted.sh</action>
          <action>FALSE recover autohealing</action>
        </rule>
      </admin_rules>
    </rules>
    <config_data>
      <configuration>
        <dst>bootstrap_config</dst>
        <variable>
          <name>TECH_PACKAGE</name>
          <val>ax</val>
        </variable>
      </configuration>
    </config_data>
```

```
  </vm_group>
</deployment>'
```

Verify VM Deployment

The following example shows how to get the operational data for a VM deployment using the command

```
show vm_lifecycle opdata tenants tenant admin deployments
<deployment_name>/<deployment_id>/<vmgroup_name>
```

```
nfvis# show vm_lifecycle opdata tenants tenant admin deployments ROUTER
deployments ROUTER
 deployment_id SystemAdminTenantIdROUTER
 vm_group ROUTER
  bootup_time 600
  vm_instance d1c462e9-2706-4868-befd-d8f7806b9444
   name     ROUTER
   host_id  NFVIS
   hostname nfvis
   interfaces interface 0
    model       virtio
    type        virtual
    port_id     vnic0
    network     int-mgmt-net
    subnet      N/A
    ip_address  10.20.0.2
    mac_address 52:54:00:fe:34:53
    netmask     255.255.255.0
    gateway     10.20.0.1
   interfaces interface 1
    type        virtual
    port_id     vnic1
    network     GE0-1-SRIOV-1
    subnet      N/A
    mac_address 52:54:00:e4:13:67
 state_machine state SERVICE_ACTIVE_STATE
VM
NAME    STATE
------------------------
ROUTER  VM_ALIVE_STATE
```

VM Deployment Parameters

VNFs can be deployed using multiple mandatory and option parameters. The following table lists some of the parameters.

Parameter	Notes	Example
`image`	Mandatory. The image needs to be registered and active when it is being referred in deployment.	`<image>ISR_IMAGE</image>`

`bootup_time`	This parameter is no longer mandatory from 3.12 and later, provided that it is specified in image properties. **Accepted Values:** • For unmonitored VMs: -1 • For monitored VMs: Number of seconds	`<bootup_time>500</bootup_time>`
`vim_vm_name`	Optional. If a custom VM name is provided at the time of deployment, it may be used for all commands that accept VM name as an input.	
`kpi_data`	Mandatory for monitored VMs.	
Rules	Mandatory for monitored VMs.	
`config_data`	Mandatory if the day-0 configuration has variables that have tokens assigned to them.	
`Encrypted config variable`	Optional. Only one value for a variable is allowed to be encrypted.	<variable> <name>TEST_VARIABLE</name> <encrypted_val>test_value</encrypted_val> </variable>

`placement`	Optional.
	The ***placement*** tag under vm group points to the location where the VNF would be deployed. This parameter supports deploying a VNF in a local data store (default-if not specified), external data store (datastore2), or NFS.
		`<placement>`
		`<type>zone_host</type>`
		`<host>nfs:nfs_storage</host>`
		`</placement>`

`volumes`	Optional. Up to 2 volumes could be added to a deployment. Location of the volumes can be local or NFS (needs NFS mount name to be specified in case of NFS)	

port_forwarding	Optional. If port forwarding is included, all elements under it are mandatory.	<port_forwarding> <port> <type>ssh</type> <protocol>tcp</protocol> <vnf_port>22</vnf_port> <external_port_range> <start>20022</start> <end>20022</end> </external_port_range> </port> </port_forwarding>
Ngio interface	Optional. Used in config_data. To enable NIM support on a Cisco ISRv running on Cisco ENCS, you must use the variables in the ISRv deployment payload.	<variable> <name>ngio</name> <val>enable</val> </variable>
Interface model	Optional. If the model is not specified for an interface, the default model is used. For Windows, the default model is rtl8139.	<interface> <nicid>3</nicid> <network>wan-net</network> <model>virtio</model> </interface>
VNC password	Optional. If the VNC password is not specified, there is no default password.	

VM Bootstrap Configuration Options with a VM Deployment

You can include the bootstrap configuration (day zero configuration) of a VM in the VM deployment payload in the following ways:

- Bundle bootstrap configuration files into the VM package: In this method, the bootstrap configuration variables can be assigned tokens. Token names must be in bold text. For each variable with a token assigned, key-value pairs must be provided during deployment in the deployment payload.
- Bootstrap configuration as part of the deployment payload: The entire bootstrap configuration is copied to the payload without tokens.
- Bootstrap configuration file in the NFVIS server: In this method, the configuration file is copied or downloaded to the NFVIS server, and referenced from the deployment payload with the filename, which includes the full path.

For examples on how to use bootstrap configuration options in the deployment payload, see the API Reference for Cisco Enterprise Network Function Virtualization Infrastructure Software.

VM Monitoring

After VMs are deployed, they monitored periodically based on the metrics defined in the KPI section of deployment data model. Monitoring can be enabled or disabled by modifying the <actionType> tag. See the VM Actions section for details on the allowed values for the action Type tag and what they mean.

The following example shows how to disable monitoring for a VM.

```
curl -k -v -u "admin:password" -H
"Accept:application/vnd.yang.data+xml" -H
"Content-Type:application/vnd.yang.data+xml" -X POST
https://<NFVIS_IP>/api/operations/vmAction --data '<vmAction>
<actionType>DISABLE_MONITOR</actionType><vmName><vm-instance name></vmName></vmAction>'
```

- If the bootup_time is set at -1, it signifies that VM monitoring is disabled.
- You are not required to set a boot up time during image registration. However, you must set it during VM deployment.
- If a qcow2 image is used during registration, the bootup_time defaults to -1.

VNF Deployment Placement

Cisco NFVIS allows you to specify where a VNF should be deployed using the placement tag for parameter deployment. See VM Deployment Parameters for more information on supported placement parameters and their accepted values.

Note If you are placing the VNF deployment on **nfs**, ensure that you have configured this storage option to be mounted on NFVIS using appropriate CLIs before deploying the VNF.

Example: VM Deployment Using Placement

```
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/vm_lifecycle/tenants
/tenant/admin/deployments --data
'<deployment>
<name>WINIsodep</name>
<vm_group>
  <name>WINIsovmgrp</name>
  <image>WinServer2012R2.iso</image>
  <flavor>windows</flavor>
  <bootup_time>-1</bootup_time>
  <recovery_wait_time>0</recovery_wait_time>
  <kpi_data>
   <enabled>true</enabled>
  </kpi_data>
   <scaling>
   <min_active>1</min_active>
   <max_active>1</max_active>
   <elastic>true</elastic>
   </scaling>
   <placement>
   <type>zone_host</type>
```

```
   <enforcement>strict</enforcement>
   <host>datastore1</host>
   </placement>
   <recovery_policy>
   <recovery_type>AUTO</recovery_type>
   <action_on_recovery>REBOOT_ONLY</action_on_recovery>
   </recovery_policy>
</vm_group>
</deployment>'
```

VNF Volumes

A VNF can be created and deployed with multiple volumes. Currently, NFVIS supports a maximum of two volumes per VNF.

Restrictions for VNF Volumes

- Volumes cannot be updated for a VNF after a VNF has already been deployed.
- NFVIS currently supports only two volumes per VNF.

Example: Payload for Creating Volumes

```
…
<volumes>
  <volume>
    <name>Volume1</name>
    <volid>1</volid>
    <bus>ide</bus>
    <size>1</size>
    <sizeunit>GiB</sizeunit>
    <format>qcow2</format>
    <device_type>disk</device_type>
    <storage_location>local</storage_location>
  </volume>
</volumes>
…
```

Volume Storage Locations

The following are the accepted values for **storage_location** tags:

- <**storage_location**>*local*</**storage_location**>
- <**storage_location**>*nfs:NFS_MOUNT_NAME*</**storage_location**>

Note The storage locations *datastore1*, *datastore2* and *datastore3* are not supported for NFVIS 3.12.3 and later releases.

Volume Deployment Parameters

Cisco NFVIS supports the following fields for volume deployment.

Property	Allowed Values	Description

name	string, { length 1..255 }	Name of the volume
volid	uint16	Volumes will be presented to the VM sorted by volume ID.
bus	ide, scsi, virtio	The bus type
size	unit 16	Size of the Volume
sizeunit	MiB,GiB,TiB,PiB,EiB	Size units. MiB/GiB/TiB/PiB/EiB
format	qcow2, raw	Format of the disk to be created.
device_type	disk, CD ROM	Type of the device being attached to the VM.
storage_location	local, nfs:NFS_MOUNT_NAME	Storage location name

Port Forwarding

By default, the wan bridge interface (**wan-br**) is used to redirect incoming traffic from WAN to access the internal management network (**int-mgmt-net**) of the VM.

The bridge interface that is used to redirect traffic coming from the WAN side can be modified using the **source_bridge** tag in the deployment payload as shown below.

```
<port_forwarding>
  <port>
  <type>ssh</type>
  <protocol>tcp</protocol>
  <vnf_port>22</vnf_port>
  <source_bridge>MGMT</source_bridge>
  <external_port_range>
    <start>20122</start>
    <end>20122</end>
  </external_port_range>
  </port>
</port_forwarding>
```

With the payload above, the traffic coming from the WAN side is redirected through the management interface **(MGMT)** instead of the default WAN bridge **(wan-br)** interface.

NGIO

Next Generation Input/Output (NGIO) is supported on Cisco ENCS platforms from NFVIS release 3.11 and later. Using the NGIO flag, the VM image informs its capability. Between ISRv and NFVIS image, NGIO is used to decide the NIM enablement capability available for the VM. This is only for Cisco ISRv with Cisco IOS XE image or Cisco IOS XE with SD-WAN image.

To enable NGIO on a VNF. a config data variable is added to the deployment payload as shown below.

```
<config_data>
  <configuration>
    <dst>bootstrap_config</dst>
    <variable>
      <name>TECH_PACKAGE</name>
      <val>ax</val>
    </variable>
    <variable>
      <name>ngio</name>
      <val>enable</val>
```

```
    </variable>
  </configuration>
```

The following is an example of the interfaces added to a VNF, when the NGIO variable is detected and set to **enable**.

```
<interface type='bridge'>
  <source bridge='csxbr'/>
  <model type='e1000'/>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0' multifunction='on'
 />
</interface>
<interface type='hostdev' managed='yes'>
  <driver name='vfio'/>
  <source>
    <address type='pci' domain='0x0000' bus='0x0e' slot='0x10' function='0x1'/>
  </source>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x1'
multifunction='on'/>
</interface>
<interface type='hostdev' managed='yes'>
  <driver name='vfio'/>
  <source>
    <address type='pci' domain='0x0000' bus='0x0e' slot='0x10' function='0x3'/>
  </source>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x2' multifunction='on'
 />
</interface>
```

Requirements for NGIO Validation

When NGO deployment request is initiated, it is validated against the specified image and the platform itself. To be accepted, the platform must be supported and the registered image must have the following properties.

```
<image_properties>
    ...
    <custom_property>
        <ngio>enable</ngio>
    </custom_property>
    ...
</image_properties>
```

VM States

The following table describes various VM states.

VM States	Description
VM_UNDEF_STATE	The initial state of a VM or VNF before deployment of this VM.
VM_DEPLOYING_STATE	The VM or VNF is being deployed on to the NFVIS.
VM_MONITOR_UNSET_STATE	The VM or VNF is deployed in NFVIS but the monitoring rules are not applied.
VM_MONITOR_DISABLED_STATE	Due to a VM action request or recovery workflow, the monitoring or KPI rules applied to the VM were not enabled.
VM_STOPPING_STATE	VM or VNF is being stopped.
VM_SHUTOFF_STATE	VM or VNF is in stopped or shutoff state.

VM States	Description
VM_STARTING_STATE	VM or VNF is being started.
VM_REBOOTING_STAT	VM or VNF is being rebooted.
VM_INERT_STATE	VM or VNF is deployed but not alive. The KPI monitor is applied and waiting for the VM to become alive.
VM_ALIVE_STATE	VM or VNF is deployed and successfully booted up or alive as shown in the KPI metric.
VM_UNDEPLOYING_STATE	The deployment of a VM or VNF is being terminated.
VM_ERROR_STATE	The VM or VNF is in an error state because the deployment or some other operation has failed.

VNF Deployment Update

After you have deployed a VNF, you can update it in terms of its flavor, CPU topology, or interfaces.

Update VNF Flavor

You can update a VNF deployment to have a different flavor from the one you deployed it with. The flavor can also be custom-defined.

Note Before updating a VNF with another flavor, we recommend that you check whether CPUs are avaialble for the required update.

Updating a VNF flavor only supports CPU and Memory changes and does not support disk size change.

The following is a typical workflow for updating a VNF flavor.

Get Available Flavors

Use the following command to get a list of available flavors.

```
curl -k -v -u admin:admin -X GET
https://209.165.201.1/api/operational/vm_lifecycle/flavors?deep
```

Check System CPU Usage

Use the following command to check the CPU usage of the system.

```
curl --tlsv1.2 -k -i -u admin:Esc123# -H
Accept:application/vnd.yang.data+json -H content-
type:application/vnd.yang.data+json -X GET
https://<nfvis_ip>/api/operational/resources/cpu-info/allocation
```

Update VNF Flavor

Use the following command to update the flavor of a VNF.

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

```
https://<nfvis_ip>/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/<deploymentID>
/vm_group/<VMGroupName>/flavor
--data
'<flavor><FlavorName></flavor>
```

Example: Changing the Flavor of a VNF from Flavor from ASAv5 to ASAv10

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H Content-Type:application/vnd.yang.data+xml -X PUT
https://172.29.91.32/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/ASAdep/vm_group/ASAvmgrp/
flavor --data '<flavor>ASAv10</flavor>
```

Update CPU Topology

Updating the CPU topology of a VNF involves updating a VM to a custom-defined topology that you would have created at the time of creating a VM flavor. For more details, see VM Profiles or Flavors, on page 97.

The process to update the CPU topology is similar to updating a VNF flavor—by replacing the name of the CPU topology. The following command is used to update a CPU topology.

```
curl -k -v -u admin:admin -H
Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml
-X PUT
https://<nfvis_ip>/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/<deploymentID>/vm_group/
<VMGroupName>/flavor
--data
'<flavor><FlavorName_withCPUTopology></flavor>'
```

Example: Updating ISRv to Include a CPU Topology

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H Content-
Type:application/vnd.yang.data+xml -X PUT
https://172.29.91.32/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/ROUTER/vm_group/ROUTER/flavor

--data '<flavor>Isrv_CPUToplogy</flavor>'
```

Verify Changed Configuration

Use the `support virsh dumpxml <uid>` command to verify whether the changed configuration was applied to the VNF.

The following is an example of the output you would see.

```
<vcpu placement='static'>3</vcpu>
  <cputune>
    <vcpupin vcpu='0' cpuset='11'/>
    <vcpupin vcpu='1' cpuset='10'/>
    <vcpupin vcpu='2' cpuset='9'/>
    <emulatorpin cpuset='21-23'/>
  </cputune>
  . . .
  <cpu mode='host-passthrough' check='none'>
    <topology sockets='1' cores='3' threads='1'/>
  </cpu>
```

About Updating VNF Interfaces

NFVIS supports updating VNF interfaces at the time of updating a VM deployment. VM interface updates include adding an interface, deleting an interface, and moving a VNIC of a VM from one interface to another.

Hot and Cold Updates

All the options related to updating interfaces mentioned above can be done in two ways: hot or cold.

Hot Update: Hot update refers to an update operation that runs when a VM is in ACTIVE state. In such cases, the VM does not reboot during the update.

Cold Update: Cold update refers to an update operation that runs after a VM is put in a VM_SHUTOFF_STATE. In such cases, the VM reboots during the update.

Note Whether you would need to do a hot or cold update depends on the custom image properties set during image registration. Refer to the table below for various custom image properties related to hot and cold updates.

Custom Image Property	Value	Interface Update Description	Default Value
interface_hot_add	true	Hot add an interface to a VM	true
interface_hot_add	false	Cold add an interface to VM	
interface_hot_delete	true	Hot delete a VM interface	false
interface_hot_delete	false	Cold delete a VM interface	

Starting from ISRv 17.1, interface_hot_add and interface_hot_delete are set to true by default.

Prerequisites for Updating VNF Interfaces

- The VNF should support hot add or hot delete operations for the interface.
- The custom image properties for interface update should be set during image registration to allow values other than the default.
- The VM to be updated needs to be in one of the following states: VM_ALIVE_STATE, VM_ERROR_STATE or VM_SHUTOFF_STATE.

Supported Interface Update Operations by Interface Types

The following table shows which hot interface update operations are supported for various interface types.

Interface Type	Hot Add	Hot Delete	Hot Update for Moving VNIC
VIRTIO	Yes	Yes	Yes
SRIOV	Yes	Yes	Yes
DPDK	Yes	Yes	Yes

Note NFVIS also supports moving VNICs from one interface to another. For example, you can move a VNIC from a VIRTIO interface to SRIOV, or from SRIOV to DPDK, and so on.

If the VNIC is updated to a different interface type like SRIOV or DPDK, the configuration of the vnic will not be preserved.

Syslog is not generated in ISRv when an interface is updated from a DPDK enabled network to another DPDK enabled network.

Update Interfaces

This topic walks you through how to perform various tasks related to updating interfaces such as adding an interface, deleting an interface, and so on.

Add Interfaces

Single Interface: The following example shows how to add a single interface to a VM deployment.

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H Content-Type:application/vnd.yang.data+xml -X PUT
https://<NfvisIpAddress>/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/ASAdep/vm_group/ASAvmgrp/interfaces
 --data '
<interfaces>
<interface>
        <nicid>0</nicid>
        <network>int-mgmt-net</network>
    </interface>
    <interface>
        <nicid>newNIC</nicid>
        <network>networkName</network>
    </interface>
    </interfaces>'
```

Multiple Interfaces: The following example shows how to add multiple interfaces (in this case, two) to a VM deployment.

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X PUT
https://172.29.91.32/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/ASAdep/vm_group/ASAvmgrp/interfaces
 --data '
<interfaces>
<interface>
          <nicid>0</nicid>
          <network>int-mgmt-net</network>
      </interface>
      <interface>
          <nicid>1</nicid>
          <network>wan-net</network>
      </interface>
      <interface>
      <nicid>2</nicid>
      <network>lan-net</network>
      </interface>
      </interfaces>'

HTTP/1.1 204 No Content
```

Delete Interfaces

The following REST API is used to delete an interface from a VM deployment.

```
curl -k -v -u admin:<password> -H
Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X PUT
https://<NfvisIpAddress>/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/ASAdep/vm_group/ASAvmgrp/interfaces
 --data '
<interfaces>
<interface>
        <nicid>0</nicid>
        <network>int-mgmt-net</network>
    </interface>
**** Note: Remove the required nicID along with content between <interface> and </interface>
 ******
    </interfaces>'
```

Example: The following example shows how to delete an interface called NIC ID2, which was added in the example of adding multiple interfaces in the Add Interfaces section above.

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X PUT
https://172.29.91.32/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/ASAdep/vm_group/ASAvmgrp/interfaces
 --data '
    <interfaces>
      <interface>
              <nicid>0</nicid>
              <network>int-mgmt-net</network>
          </interface>
          <interface>
              <nicid>1</nicid>
              <network>wan-net</network>
       </interface>
     </interfaces>'

HTTP/1.1 204 No Content
```

Notice that in the example above, NIC ID 2 has been excluded from the REST API for it to be deleted from the deployment.

Move VNICs from One Network to Another

```
curl -k -v -u admin:<password> -H
Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X PUT
https://<NfvisIpAddress>/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/ASAdep/vm_group/ASAvmgrp/interfaces
 --data '
<interfaces>
    <interface>
        <nicid>0</nicid>
        <network>int-mgmt-net</network>
    </interface>
    <interface>
        <nicid>selectedNicId</nicid>
         <network>NewNetworkSelected</network>
    </interface>
</interfaces>'
```

Example: The following example shows how to move nicid 1 from wan-net to wan2-net .

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H Content-Type:application/vnd.yang.data+xml -X PUT
https://172.29.91.32/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/ASAdep/vm_group/ASAvmgrp/interfaces
```

```
 --data '
<interfaces>
         <interface>
                 <nicid>0</nicid>
                 <network>int-mgmt-net</network>
             </interface>
             <interface>
                 <nicid>1</nicid>
                 <network>wan2-net</network>
          </interface>
</interfaces>'

HTTP/1.1 204 No Content
```

Access VNFs

In Cisco NFVIS, you can access VNFs in two ways after they have been deployed: through the VNC Console and through the Serial Console.

Access VNFs Using VNC Console

A VNC console allows you to access VNFs through a VNC client. This method enables you to manipulate the VNF through the NFVIS portal.

If you are using the NFVIS portal, follow these steps to access the VNF using the VNC console.

1. On the NFVIS portal, click the **VM Life Cycle** tab.
2. Next, click the black console icon next to the VNF you want to access (see screenshot below).

Secure Access to the VNC Console

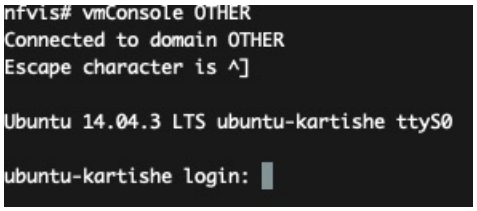
For added security, access to the VNC console can be restricted through a passphrase. To enable VNC passphrase, include the contents as shown in the example below.

```
<vm_lifecycle>
 <tenants>
  <name>admin</name>
   <deployments>
    <deployment>
     …
     <vnc>
      <password>PASSWORD</password>
     </vnc>
    </deployment>
   </deployments>
  </tenant>
 </tenants>
</vm_lifecycle>
```

Access VMs Using Serial Console

The serial console allows you to access the VM using the serial interface provided by the VM itself. This method is applicable only if the VM supports serial interfaces in both, its image and its image properties.

To access the VNF through the serial console, use the command `vmConsole` followed by the name of the VNF as shown in the CLI example below.



Access a VM Using Port Forwarding

To access a VM using port forwarding:

1. Deploy a VNF using the deployment payload with **port_forwarding** configuration:

```
<port_forwarding>
  <port>
    <type>ssh</type>
    <protocol>tcp</protocol>
    <vnf_port>22</vnf_port>
    <external_port_range>
      <start>20122</start>
      <end>20122</end>
    </external_port_range>
  </port>
  <port>
    <type>telnet</type>
    <protocol>tcp</protocol>
    <vnf_port>23</vnf_port>
    <external_port_range>
      <start>20123</start>
      <end>20123</end>
    </external_port_range>
  </port>
</port_forwarding>
```

2. Log into VNF using SSH and port number given in the example payload (20122):

```
USER-M-G2PT:~ user$ ssh cisco@172.29.91.28 -p 20122
Password:

isrv-encs#
```

Import and Export NFVIS VM

Starting from NFVIS 3.10.1 release, you can backup or export (vmExportAction) and restore or import (vmImportAction) VMs. To backup or restore the whole NFVIS system, refer Backup and Restore NFVIS and VM Configurations.

VM Export and Import Limitations

- The imported VM cannot change datastore.
- The original registered image must exist.
- The OVS network name must be identical to the one used by original deployment.
- NFVIS does not check the disk space before exporting or importing a VM.

To export a VM ensure that:

- VM is in powered off state.
- Backup file must be saved to NFVIS datastore or USB.
- Provide a backup name for NFVIS to append .vmbkp extension to the backup name.

You can only create and save a VM backup to datastores. The backup file has .vmbkp extension. To verify the backup:

```
nfvis# show system file-list disk local | display xpath | include backup

/system/file-list/disk/local[si-no='84']/name tiny_backup.vmbkp
nfvis# show system file-list disk local 84
SI NO  NAME               PATH                SIZE  TYPE               DATE MODIFIED
-------------------------------------------------------------------------------------------
84     tiny_backup.vmbkp  /mnt/extdatastore1  17M   VM Backup Package  2019-01-31 19:31:32
```

To import a VM ensure that:

- The Backup file is placed under NFVIS datastores or USB.
- The registered image used by the original deployed VM is in the same datastore, with same properties.
- The exported VM does not exist on the system.
- OVS network used by the original deployment should exist.
- Restored VM is created with the same datastore with same deployment properties.
- The full path name to backup file is used (for example, /mnt/extdatastore1/backup.vmbkp, not extdatastore1:backup)

```
nfvis# vmImportAction importPath /mnt/extdatastore1/tiny_backup.vmbkp
System message at 2019-01-31 19:53:32...
Commit performed by admin via ssh using maapi.
```

The following examples show export failures:

- Original deployment is not deleted

```
nfvis# vmImportAction importPath /mnt/extdatastore1/tiny_backup.vmbkp
Error: Exception from action callback: Deployment Configuration :
'SystemAdminTenantIdtiny' already exists , can not be imported/restored due to conflict!
```

- 2. OVS network used by original deployment is deleted.

```
nfvis# vmImportAction importPath /mnt/extdatastore1/tiny_backup.vmbkp
Error: Exception from action callback: Restoration Request rejected, see logs for root
 cause
```

Secure Boot of VNFs

A key aspect of a secure compute system is to ensure that the system is running the intended software without malware or tampered software. This protection must begin as soon as the system is powered-on. The UEFI (Unified Extensible Firmware Interface) specification defines a secure boot methodology that prevents loading software which is not signed with an acceptable digital signature.

NFVIS already supports UEFI Secure Boot of Host. Secure boot ensures that the booted NFVIS software is genuine. Starting from NFVIS 3.11.1 release, the UEFI secure boot feature has been extended to VNFs being used in the Cisco NFV solution.

Booting a VNF securely requires the environment to support UEFI secure boot and requires modification to VNFs to support secure boot. NFVIS 3.11.1 and later releases have the infrastructure to support UEFI secure boot for VNFs that are capable of secure boot.

Note ISRv versions 16.12 and later support secure boot.

The firmware modes in which VNFs can boot are:

- BIOS - for VNFs which are not capable of secure boot
- UEFI secure - for VNFs capable of secure boot

VNFs can indicate secure boot capability using properties in the image_properties.xml file in the tar.gz package for the VNF.

You must set the following properties to enable secure boot of VNFs:

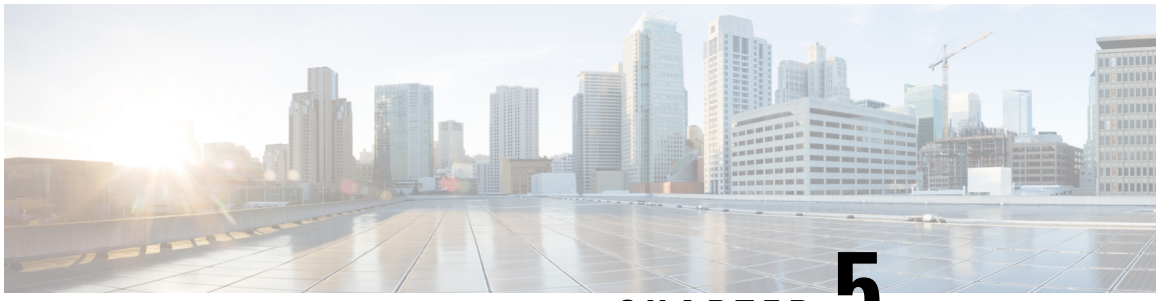
Name of the Image Property	Value for non-secure boot VNF	Value for secure boot VNF
boot_mode	bios	efi-secure-boot
shim_signature	N/A	microsoft **Note** For secure-boot capable VNFs, the VNF shim is typically signed by Microsoft.

Any combinations not matching the above default to the following:

- boot_mode: bios

- shim_signature: N/A

Note On the NFVIS portal, the **Image Repository** page shows if the image is capable of secure boot.



CHAPTER 5

Secure Overlay and Single IP Configuration

Secure Overlay

An overlay is a virtualized network layer on top of the physical network with the support of its infrastructure to provide additional security to the newtork. IPSec is a framework with protocols and algorithms to provide secured data transimission over unprotected or untrusted networks. IPSec secure tunnel is created between two networks to ensure virtual private network communication.

Secure overlay in NFVIS allows IPSec tunnel establishment between NFVIS supporting the vBranch platform and a VPN server and allows the orchestrator to manage NFVIS over the IPSec tunnel.

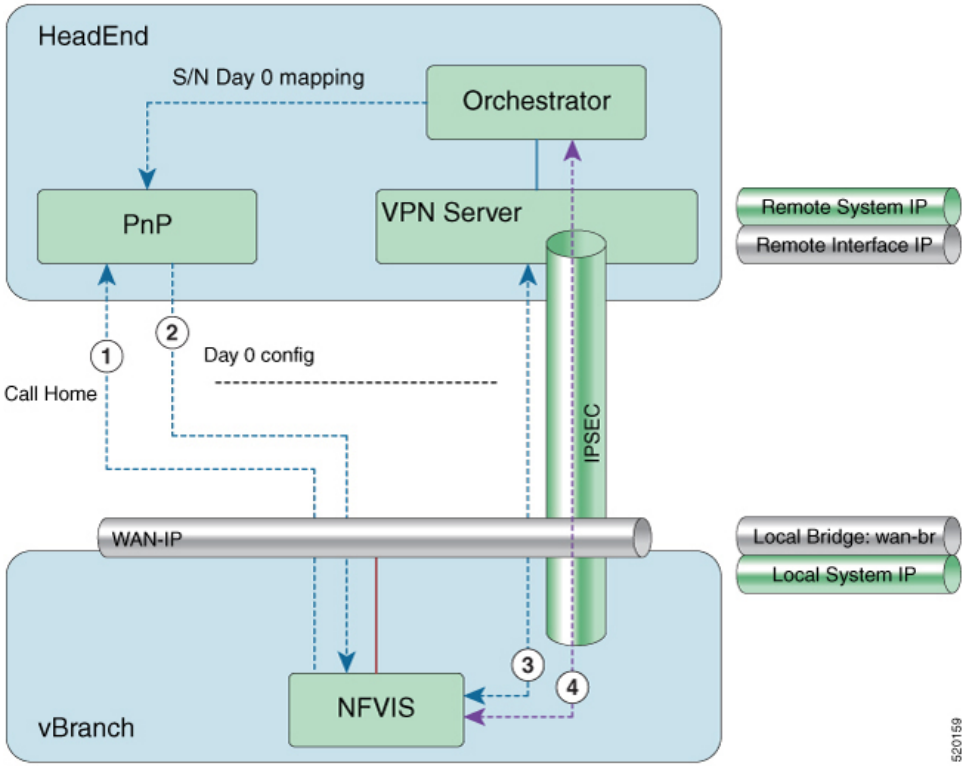
Supported Features on Secure Overlay

The following features are supported on NFVIS 3.10.x and later releases:

- IPSec IKEv2
- IPv4
- Authentication:
  - Pre-shared-key authentication
  - Introduced in NFVIS 3.12.3 release - EAP authentication
- IKE cipher:
  - aes128-sha1-mopd1536
  - Introduced in NFVIS 3.12.3 release - aes256-sha512-modp2048
  - Introduced in NFVIS 3.12.3 release - aes256-sha512-modp4096
- ESP cipher:
  - aes128-sha1
  - Introduced in NFVIS 3.12.3 release - aes256-sha512

- Local system IP address:
  - Unique tunnel IP address for each NFVIS system.
  - Introduced in NFVIS 3.11.1 release - Internal management network bridge (int-mgmt-net-br) gateway IP address is allowed to be used as local system IP address. In this case, the local system IP bridge much be set to internal management network (int-mgmt-net).
- Local bridge for NFVIS reaching out to remote VPN server:
  - wan-br by default
  - wan2-br
- Introduced in NFVIS 3.12.1 release - Secure overlay is support on NFVIS Dual WAN feature. DHCP client toggles between wan and wan2 to request for an IP address. When IP address and default gateway are obtained from an interface with DHCP configuration, the toggling stops. If dual-local-bridge is configured, to start overlay, NFVIS selects the interface between local-bridge and dual-local-bridge, in the following order:
  - Interface with DHCP configuration.
  - Interface having static IP address.
  - If both interfaces have static IP address, local-bridge interface.
- Local identity:
  - IP address or FQDN
  - Introduced in NFVIS 3.12.3 release - email domain
- Remote identity:
  - IP address or FQDN
  - Introduced in NFVIS 3.12.3 release - Distinguish Name
  - Introduced in NFVIS 3.12.3 release - email domain

Example for Secure Overlay with Zero Touch Deployment



1. NFVIS has WAN IP address, static IP address or DHCP IP address. NFVIS calls home PnP server.
2. The PnP server pushes NFVIS Day-0 configurations including the secure overlay configuration.
3. NFVIS establishes IPSec connection between NFVIS and the headend management hub which has IPSec VPN configurations. On NFVIS side, the tunnel end point has NFVIS local system IP address.
4. After the IPSec tunnel is up, the headend can connect to NFVIS through the system IP address and manage NFVIS over the IPSec tunnel.

To configure secure overlay:

```
configure terminal
secure-overlay mgmthub
remote-interface-ip-addr 10.85.189.36
    local-bridge wan-br
    remote-system-ip-addr 10.19.18.251
    remote-id mgmt-hub.cloudvpn.com
    local-system-ip-addr 14.14.14.4
    psk local-psk Cisco1234Admin
    remote-psk Cisco1234Admin
    commit


confirgure terminal
secure-overlay myconn
local-system-ip-addr 12.12.12.1
local-system-ip-bridge int-mgmt-net
```

```
remote-interface-ip-addr 172.19.160.75
    remote-system-ip-addr 192.168.1.90
    ike-cipher aes256-sha512-modp2048
    esp-cipher aes256-sha512
    remote-id "CN=vbranch, unstructuredAddress=172.19.160.75,
unstructuredName=Headend.headendvpn"
    local-id AxxxY@cisco.com
    commit


configure terminal
secure-overlay myconn eap
username admin
password Cisco123#
cacert intdatastore:uploads/csr.pem
commit
```

To get the secure overlay state:

```
nfvis# show secure-overlay
                ACTIVE                  SELECTED
                LOCAL       STATE       LOCAL
NAME    STATE   BRIDGE      DETAILS     BRIDGE
-----------------------------------------------------------
MYCONN   UP     wan-br                  wan-br
```

Examples for Configuring Secure Overlay

Note Secure overlay configuration on NFVIS must match with VPN configuration on the VPN server. The secure overlay tunnel will not be established successfully if the configurations do not match.

Secure Overlay over WAN with pre-shared-key and fqdn-remote-id

```
<secure-overlay>
   <name>mgmthub</name>
   <local-bridge>wan-br</local-bridge>
   <local-system-ip-addr>14.14.14.4</local-system-ip-addr>
   <remote-interface-ip-addr>10.85.189.36</remote-interface-ip-addr>
   <remote-system-ip-addr>10.19.18.251</remote-system-ip-addr>
   <remote-id>mgmt-hub.cloudvpn.com</remote-id>
   <psk>
      <local-psk>Cisco1234Admin</local-psk>
      <remote-psk>Cisco1234Admin</remote-psk>
   </psk>
</secure-overlay>
```

VPN configuration on VPN server:

```
crypto ikev2 authorization policy default
 route set interface
 route set access-list Inject

crypto ikev2 profile default
 match identity remote any
 identity local fqdn mgmt-hub.cloudvpn.com
```

```
 authentication local pre-share key Cisco1234Admin
 authentication remote pre-share key Cisco1234Admin
 dpd 60 2 on-demand
 nat keepalive 25
 aaa authorization group psk list default default
 virtual-template 1

crypto ipsec transform-set NO-ENCR esp-aes esp-sha-hmac
 mode tunnel

crypto ipsec profile default
 set transform-set NO-ENCR
 set ikev2-profile default

interface Loopback1
 description for IKEv2
 ip address 10.253.254.1 255.255.255.255

interface GigabitEthernet0/0/1
 description Corp_Network
 ip address 10.85.189.36 255.255.255.0
 negotiation auto

interface GigabitEthernet0/0/2
 ip address 10.19.18.250 255.255.255.0
 negotiation auto

interface Virtual-Template1 type tunnel
 ip unnumbered Loopback1
 ip mtu 1400
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet0/0/1
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile default

ip access-list extended Inject
 remark restricts customer mgmt addresses
 permit ip 10.254.0.0 0.0.255.255 any
```

Internal management network bridge IP address as local system IP address

Note NFVIS internal management network has gateway IP address 12.12.12.1.

```
<secure-overlay>
   <name>mgmthub</name>
   <local-bridge>wan-br</local-bridge>
   <local-system-ip-addr>12.12.12.1</local-system-ip-addr>
   <local-system-ip-bridge>int-mgmt-net</local-system-ip-bridge>
   <remote-interface-ip-addr>10.85.189.36</remote-interface-ip-addr>
   <remote-system-ip-addr>10.19.18.251</remote-system-ip-addr>
   <remote-id>mgmt-hub.cloudvpn.com</remote-id>
   <psk>
      <local-psk>Cisco1234Admin</local-psk>
      <remote-psk>Cisco1234Admin</remote-psk>
   </psk>
</secure-overlay>
```

dual-local-bridge and int-mgmt-net-br IP as local system IP

```
<secure-overlay>
   <name>mgmthub</name>
   <local-bridge>wan-br</local-bridge>
   <dual-local-bridge>wan2-br</dual-local-bridge.
   <local-system-ip-addr>12.12.12.1</local-system-ip-addr>
   <local-system-ip-bridge>int-mgmt-net</local-system-ip-bridge>
   <remote-interface-ip-addr>10.85.189.36</remote-interface-ip-addr>
   <remote-system-ip-addr>10.19.18.251</remote-system-ip-addr>
   <remote-id>mgmt-hub.cloudvpn.com</remote-id>
   <psk>
      <local-psk>Cisco1234Admin</local-psk>
      <remote-psk>Cisco1234Admin</remote-psk>
   </psk>
</secure-overlay>
```

EAP authentication

```
<secure-overlay>
   <name>mgmthub</name>
   <local-bridge>wan-br</local-bridge>
   <local-system-ip-addr>12.12.12.1</local-system-ip-addr>
   <local-system-ip-bridge>int-mgmt-net</local-system-ip-bridge>
   <local-id>branch101@cisco.com</local-id>
   <remote-interface-ip-addr> 172.19.160.75</remote-interface-ip-addr>
   <remote-system-ip-addr> 192.168.1.90</remote-system-ip-addr>
   <remote-id>CN=vbranch, unstructuredAddress=172.19.160.75,
unstructuredName=Headend.headendvpn</remote-id>
   <ike-cipher>aes256-sha512-modp2048</ike-cipher>
   <esp-cipher>aes256-sha51</esp-cipher>
   <eap>
      <username>admin</username>
      <password>Cisco123#</password>
      <cacert>https://cert/csr.pem</cacert>
   </eap>
</secure-overlay>
```

The following is an example of the VPN configuration on VPN server:

```
aaa group server radius radius-group
 server-private 172.19.160.190 auth-port 1812 acct-port 1813 key Cisco123#
 ip radius source-interface GigabitEthernet

aaa authentication login default group radius-group
aaa authentication login ucpe-authen group radius-group

ip domain name headendvpn

crypto pki server ca-server
 database level names
 no database archive
 hash sha512
 lifetime certificate 3650
 lifetime ca-certificate 7305 23 59
 auto-rollover 365
 eku server-auth client-auth
 database url flash:ca

crypto pki trustpoint ca-server
 revocation-check crl
 rsakeypair ca-server
```

```
crypto pki trustpoint router
 enrollment url http://172.19.160.75:80
 ip-address 172.19.160.75
 subject-name CN=vbranch
 revocation-check crl
 rsakeypair router
 auto-enroll regenerate
 hash sha512

crypto ikev2 authorization policy uCPE-athor-pol
 pfs
 route set interface

no crypto ikev2 authorization policy default

crypto ikev2 proposal uCPE-proposal
 encryption aes-cbc-256
 integrity sha512
 group 16 14

no crypto ikev2 policy default

crypto ikev2 policy uCPE-policy
 match address local 172.19.160.75
 proposal uCPE-proposal
crypto ikev2 profile uCPE-profile
 description uCPE profile
 match identity remote email domain cisco.com
 identity local dn
 authentication local rsa-sig
 authentication remote eap query-identity
 pki trustpoint router
 dpd 60 2 on-demand
 aaa authentication eap ucpe-authen
 aaa authorization group eap list default uCPE-athor-pol
 virtual-template 1 mode auto

crypto ipsec transform-set tset_aes_256_sha512 esp-aes 256 esp-sha512-hmac
 mode tunnel

crypto ipsec profile uCPE-ips-prof
 set security-association lifetime seconds 28800
 set security-association idle-time 1800
 set transform-set tset_aes_256_sha512
 set pfs group16
 set ikev2-profile uCPE-profile

interface Loopback1
 ip address 192.168.254.1 255.255.255.0

interface GigabitEthernet1
 ip address 172.19.160.75 255.255.255.0
 negotiation auto
 no mop enabled
 no mop sysid

interface GigabitEthernet2
 ip address 192.168.1.90 255.255.255.0
 negotiation auto
 no mop enabled
 no mop sysid

interface Virtual-Template1 type tunnel
 description uCPE virt template
```

```
ip unnumbered Loopback1
ip mtu 1400
ip tcp adjust-mss 1360
tunnel source GigabitEthernet1
tunnel mode ipsec ipv4
tunnel protection ipsec profile uCPE-ips-prof
```

Single Public IP Address and Secure Overlay

Single Public IP Address

In a virtual branch deployment, two public IP addresses are needed for each branch site, one for the NFVIS hypervisor and the other one for the WAN router. In Single Public IP Address feature on NFVIS, one public IP address assigned to a branch site, is seamlessly shared between the NFVIS hypervisor and the guest VM deployed on NFVIS. This feature ensures that the branch site is reachable even if the guest router is in failure state.

NFVIS reclaims the WAN IP address if the guest router has:

- Deployment failure.
- Error state.
- Stopped.
- Undeployed.

NFVIS releases the WAN IP address if the guest router has:

- Deployed.
- Started.

To create a single-ip-mode:

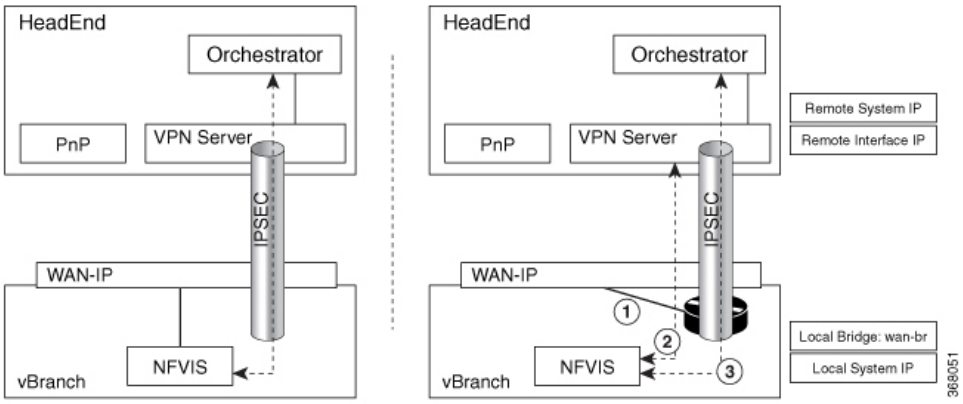
```
configure terminal
single-ip-mode vm-name ROUTER.ROUTER
commit
```

To get the state of single-ip-mode use the **show single-ip-mode** command.

Single Public IP Address with Secure Overlay

Secure overlay tunnel is established automatically when IP address is moves back and forth between NFVIS and the guest VM. The orchestrator can always reach NFVIS through the system IP address which does not change during the transitioning of the single public IP address.

Figure 2: Example of Setting IPSec Tunnel in Single IP mode



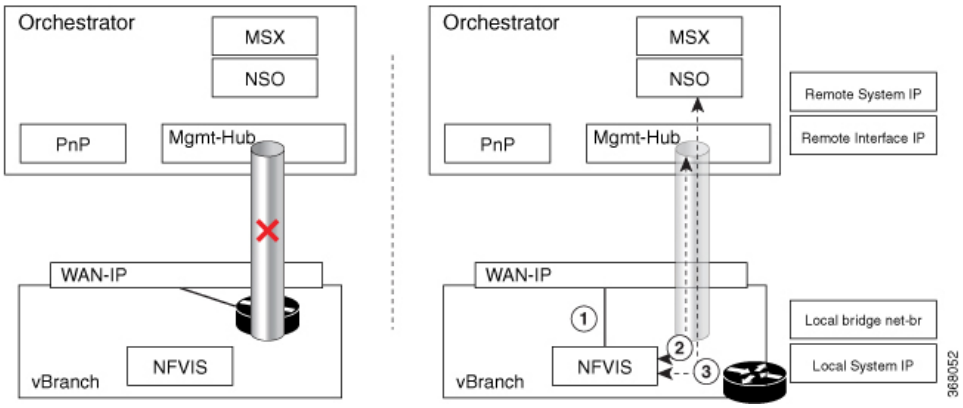
After secure overlay over WAN is established, the orchestrator sends requests to configure single IP mode and deploy the guest router that takes the public IP address.

1. NFVIS deploys the VM with specified bootstrap and Day-0 configuration. NFVIS takes down the current IPSec tunnel and releases the public IP address.
2. The VM takes the public IP address when it is in active state. NFVIS sets up the IPSec tunnel again with the remote management hub.
3. After the IPSec tunnel is up, the orchestrator can connect to NFVIS through its system IP address and manage NFVIS over the IPSec tunnel.

In single IP mode, NFVIS monitors the guest VM taking the public IP address. NFVIS takes WAN IP address back when the guest VM is:

- In error state.
- Stopped through vmAction.
- Undeployed.

Figure 3: Example of NFVIS Handling Failure



1. NFVIS takes WAN IP address.
2. NFVIS sets up IPSec tunnel to the management hub.

3. When IPSec tunnel is up, the VPN server can connect to NFVIS through its system IP address and manage NFVIS over the IPSec tunnel.

Guest VM taking Public IP Address

Guest VM must be deployed as a monitored VM which has two interfaces:

- Interface facing public with the public IP address.
- Interface on int-mgmt-net-br for traffic flow with NFVIS.

The guest VM has routing function to route traffic between the two interfaces and Network address translation (NAT) enabled. NFVIS reaches remote through int-mgmt-net-br to the guest VM.

The int-mgmt-net-br address pool and gateway IP address must be unique on each NFVIS. If secure overlay is configured, single IP mode is setup when VM is active and int-mgmt-net-br is used as a local-bridge.

Single IP address and DHCP

NFVIS single-ip-mode supports the public IP address acquired through DHCP by leveraging on the lease timer configuration on DHCP server. The guest VM with Day-0 configuration gets the IP address through DHCP when NFVIS client sends release message to DHCP server.

To handle failure, NFVIS:

- stops the VM, to ensure the VM dhclient does not send DHCP renew to DHCP server
- switches back to WAN and its dhclient sends DHCP renew message to DHCP server
- gets the same IP address from DHCP server when VM's lease time expires.

ISRv bootstrap and Day-0 Configuration

In single-ip-mode, NFVIS reaches to the guest router and takes its IP address. Traffic must be allowed between ISRv gigabit ethernet interface 1 connected to NFVIS int-mgmt-net-br and gigabit ethernet interface 2 connected to public side having the public IP address.

To verify single-ip-mode status use the **show single-ip-mode** and **show secure-overlay** command.

Single IP and Secure Overlay APIs

Secure Overlay APIs	Secure Overlay Commands
/api/config/single-ip-mode	single-ip-mode
/api/operational/single-ip-mode	

Single IP Address Without Secure Overlay

Note This feature is only supported for WAN bridge in NFVIS 3.10.1 release.

To reach NFVIS when secure overlay is not configured, you must first configure the guest device and manage IP addressing. The rest of the functionality, switching IP address between NFVIS and the guest device is the same as IP address with secure overlay.

Typically you need two IP addresses in each site, one for NFVIS and one for the VM. You can enable the single IP feature to reduce one public IP address. The single public IP address is used by NFVIS after deploying the VM with the single IP feature. After the VM comes up, NFVIS releases the public IP address for the VM to use. NFVIS and the VM have an internal network to communicate with each other. The traffic between NFVIS and an external network will need to go through the new VM and NAT by the new VM.

For single IP without secure overlay feature to work:

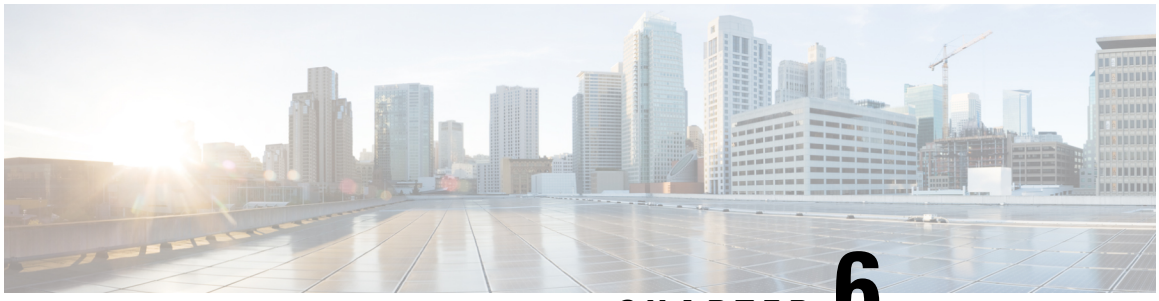
- From the **Deploy** page on NFVIS portal select single IP or configure the single IP mode by using the **single-ip-mode router.router** command.
- Provide a bootstrap file for the VM.
- Enable **Monitor** for the VM and the internal network int-mgmt-net between NFVIS and VM is created automatically.

The following example is a sample bootstrapping configuration:

172.25.221.7/24 is the single public IP address that is originally used by NFVIS and later by the VM. 172.25.221.1 is the gateway to the external network and 10.20.0.x is the internal network between NFVIS and the VM. IP address in 10.20.0.x network is used to NAT by the VM: -

```
----------------------
interface GigabitEthernet1
ip nat inside
negotiation auto
!
interface GigabitEthernet2
ip address 172.25.221.17 255.255.255.0
ip nat outside
negotiation auto
!
ip nat inside source list NAT interface GigabitEthernet2 overload
ip route 0.0.0.0 0.0.0.0 172.25.221.1
!
ip access-list standard NAT
permit 10.20.0.0 0.0.0.25
-------------
```

When the VM is down, NFVIS takes back the single IP address and the external server can communicate with NFVIS directly.

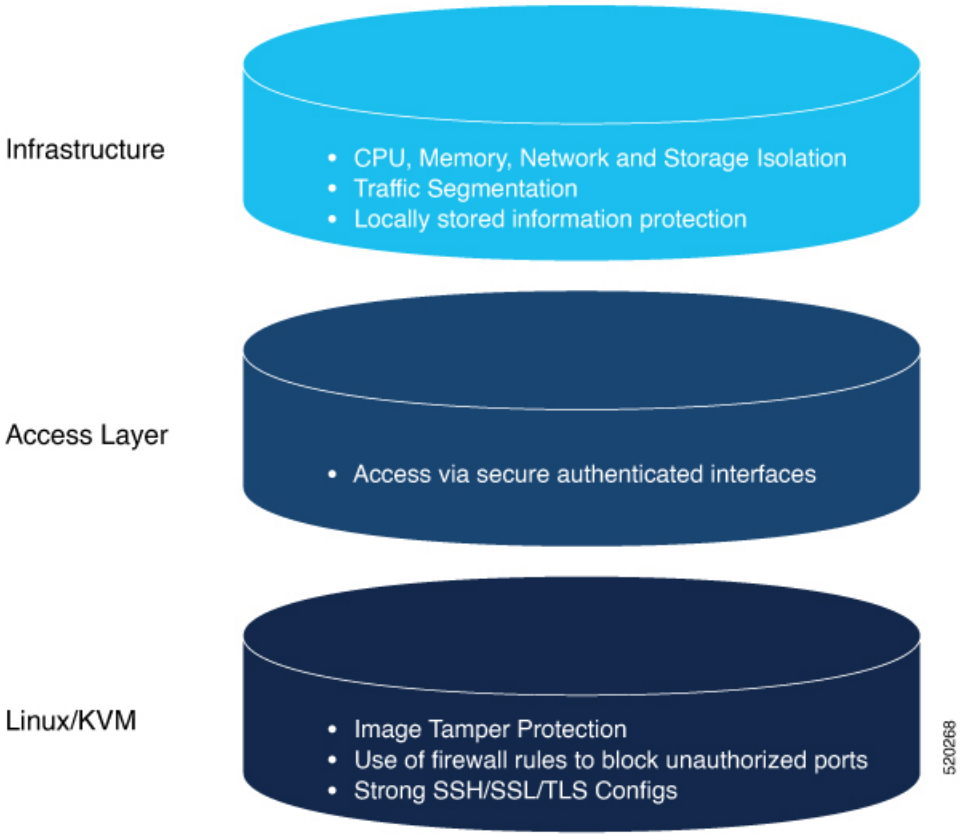


CHAPTER 6

Security Considerations

This chapter describes the security features and considerations in NFVIS. It gives a high-level overview of security related components in NFVIS to plan a security strategy for deployments specific to you. It also has recommendations on security best practices for enforcing the core elements of network security.

The NFVIS software has security embedded right from installation through all software layers. The subsequent chapters focus on these out-of-the-box security aspects such as credential management, integrity and tamper protection, session management, secure device access and more.



Installation

To ensure that the NFVIS software has not been tampered with , the software image is verified before installation using the following mechanisms:

Image Tamper Protection

NFVIS supports RPM signing and signature verification for all RPM packages in the ISO and upgrade images.

RPM Signing

All RPM packages in the Cisco Enterprise NFVIS ISO and upgrade images are signed to ensure cryptographic integrity and authenticity. This guarantees that the RPM packages have not been tampered with and the RPM packages are from NFVIS. The private key used for signing the RPM packages is created and securely maintained by Cisco.

RPM Signature Verification

NFVIS software verifies the signature of all the RPM packages before an installation or upgrade. The following table describes the Cisco Enterprise NFVIS behavior when the signature verification fails during an installation or upgrade.

Scenario	Description
Cisco Enterprise NFVIS 3.7.1 and later installations	If the signature verification fails while installing Cisco Enterprise NFVIS, the installation is aborted.
Cisco Enterprise NFVIS upgrade from 3.6.x to Release 3.7.1	The RPM signatures are verified when the upgrade is being performed. If the signature verification fails, an error is logged but the upgrade is completed.
Cisco Enterprise NFVIS upgrade from Release 3.7.1 to later releases	The RPM signatures are verified when the upgrade image is registered. If the signature verification fails, the upgrade is aborted.

Image Integrity Verification

RPM signing and signature verification can be done only for the RPM packages available in the Cisco NFVIS ISO and upgrade images. To ensure the integrity of all the additional non-RPM files available in the Cisco NFVIS ISO image, a hash of the Cisco NFVIS ISO image is published along with the image. Similarly, a hash of the Cisco NFVIS upgrade image is published along with the image. To verify that the hash of Cisco

NFVIS ISO image or upgrade image matches the hash published by Cisco, run the following command and compare the hash with the published hash:

```
% /usr/bin/sha512sum <ImageFile>
c2122783efc18b039246ae1bcd4eec4e5e027526967b5b809da5632d462dfa6724a9b20ec318c74548c6bd7e9b8217ce96b5ece93dcddf74fda5e01bb382ad607
  <ImageFile>
```

ENCS Secure Boot

Secure boot is part of the Unified Extensible Firmware Interface (**UEFI**) standard which ensures that a device boots only using a software that is trusted by the Original Equipment Manufacturer (OEM). When NFVIS starts, the firmware checks the signature of the boot software and the operating system. If the signatures are valid, the device boots, and the firmware gives the control to the operating system.

Secure boot is available on the ENCS but is disabled by default. Cisco recommends you to enable secure boot. For more information, see Overview to ENCS 5400 for UEFI Secure Boot.

Secure Unique Device Identification

NFVIS uses a mechanism known as Secure Unique Device Identification (SUDI), which provides it with an immutable identity. This identity is used to verify that the device is a genuine Cisco product, and to ensure that the device is well-known to the customer's inventory system.

The SUDI is an X.509v3 certificate and an associated key-pair which are protected in hardware. The SUDI certificate contains the product identifier and serial number and is rooted in Cisco Public Key Infrastructure. The key pair and the SUDI certificate are inserted into the hardware module during manufacturing, and the private key can never be exported.

The SUDI-based identity can be used to perform authenticated and automated configuration using Zero Touch Provisioning (ZTP). This enables secure, remote on-boarding of devices, and ensures that the orchestration server is talking to a genuine NFVIS device. A backend system can issue a challenge to the NFVIS device to validate its identity and the device will respond to the challenge using its SUDI based identity. This allows the backend system to not only verify against its inventory that the right device is in the right location but also provide encrypted configuration that can only be opened by the specific device, thereby ensuring confidentiality in transit.

The following workflow diagrams illustrate how NFVIS uses SUDI:

Figure 4: Plug and Play (PnP) Server authentication

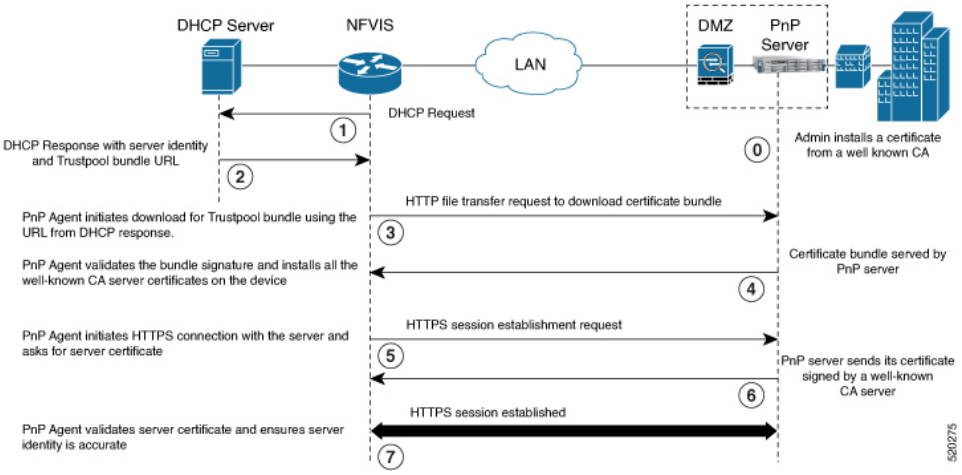
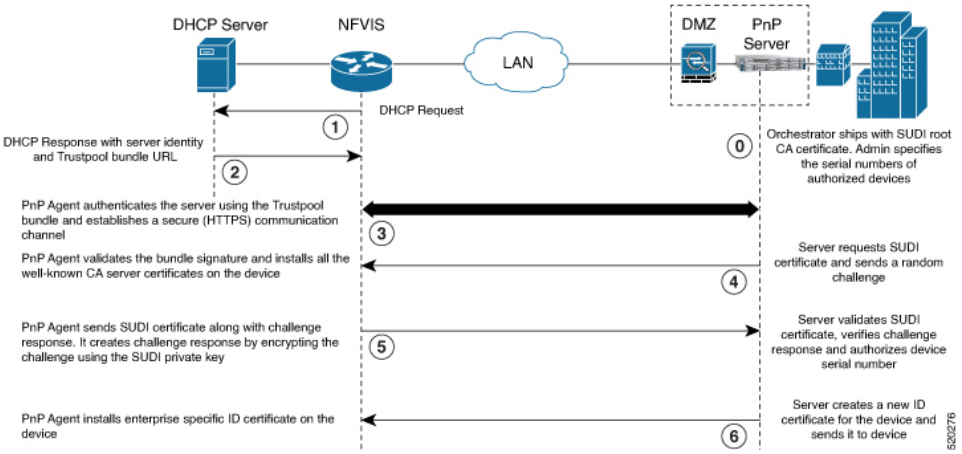


Figure 5: Plug and Play Device Authentication and Authorization



Device Access

NFVIS provides different access mechanisms including console as well as remote access based on protocols such as HTTPS and SSH. Each access mechanism should be carefully reviewed and configured. Ensure that only the required access mechanisms are enabled and that they are properly secured. The key steps to securing both interactive and management access to NFVIS are to restrict the device accessibility, restrict the capabilities of the permitted users to what is required, and restrict the permitted methods of access. NFVIS ensures that the access is only granted to authenticated users and they can perform just the authorized actions. Device access is logged for auditing and NFVIS ensures the confidentiality of locally stored sensitive data.

It is critical to establish the appropriate controls in order to prevent unauthorized access to NFVIS. The following sections describe the best practices and configurations to achieve this:

Enforced Password Change at First Login

Default credentials are a frequent source of product security incidents. Customers often forget to change the default login credentials leaving their systems open to attack. To prevent this, the NFVIS user is forced to change the password after the first login using the default credentials (username: admin and password Admin123#).

For more information, see Accessing NFVIS, on page 6.

Restricting Login Vulnerabilities

You can prevent the vulnerability to dictionary and Denial of Service (DoS) attacks by using the following features.

Enforcement of Strong password

An authentication mechanism is only as strong as its credentials. For this reason, it is important to ensure users have strong passwords. NFVIS checks that a strong password is configured as per the following rules:

Password must contain:

- At least one uppercase character
- At least one lowercase character
- At least one number
- At least one of these special characters: hash (#), underscore (_), hyphen (-), asterisk (*), or question mark (?)
- Seven characters or more
- The password length should be between 7 and 128 characters.

Configuring Minimum Length for Passwords

Lack of password complexity, particularly password length, significantly reduces the search space when attackers try to guess user passwords, making brute-force attacks much easier.

The admin user can configure the minimum length required for passwords of all users. The minimum length must be between 7 and 128 characters. By default, the minimum length required for passwords is set to 7 characters.

CLI:

```
nfvis(config)# rbac authentication min-pwd-length 9
```

API:

```
/api/config/rbac/authentication/min-pwd-length
```

Configuring Password Lifetime

The password lifetime determines how long a password can be used before the user is required to change it.

The admin user can configure minimum and maximum lifetime values for passwords for all users and enforce a rule to check these values. The default minimum lifetime value is set to 1 day and the default maximum lifetime value is set to 60 days.

When a minimum lifetime value is configured, the user cannot change the password until the specified number of days have passed. Similarly, when a maximum lifetime value is configured, a user must change the password before the specified number of days pass. If a user does not change the password and the specified number of days have passed, a notification is sent to the user.

Note The minimum and maximum lifetime values and the rule to check for these values are not applied to the admin user.

CLI:

```
configure terminal
rbac authentication password-lifetime enforce true min-days 2 max-days 30
commit
```

API:

```
/api/config/rbac/authentication/password-lifetime/
```

Limit previous password reuse

Without preventing the use of previous passphrases, password expiry is largely useless since users can simply change the passphrase and then change it back to the original.

NFVIS checks that the new password is not the same as one of the 5 previously used passwords. One exception to this rule is that the admin user can change the password to the default password even if it was one of the 5 previously used passwords.

Restrict Frequency of login attempts

If a remote peer is allowed to login an unlimited number of times, it may eventually be able to guess the login credentials by brute force. Since passphrases are often easy to guess, this is a common attack. By limiting the rate at which the peer can attempt logins, we prevent this attack. We also avoid spending the system resources on unnecessarily authenticating these brute-force login attempts which could create a Denial of Service attack.

NFVIS enforces a 5 minute user lockdown after 10 failed login attempts.

Disable inactive user accounts

Monitoring user activity and disabling unused or stale user accounts helps to secure the system from insider attacks. The unused accounts should eventually be removed.

The admin user can enforce a rule to mark unused user accounts as inactive and configure the number of days after which an unused user account is marked as inactive. Once marked as inactive, that user cannot login to the system. To allow the user to login to the system, the admin user can activate the user account.

Note The inactivity period and the rule to check the inactivity period are not applied to the admin user.

The following CLI and API can be used to configure the enforcement of account inactivity.

CLI:

```
configure terminal
rbac authentication account-inactivity enforce true inactivity-days 30
commit
```

API:

```
/api/config/rbac/authentication/account-inactivity/
```

The default value for inactivity-days is 35.

Activating an Inactive User Account

The admin user can activate the account of an inactive user using the following CLI and API:

CLI:

```
configure terminal
rbac authentication users user guest_user activate
commit
```

API:

```
/api/operations/rbac/authentication/users/user/username/activate
```

Integration with external AAA servers

Users login to NFVIS through ssh or the Web UI. In either case, users need to be authenticated. That is, a user needs to present password credentials in order to gain access.

Once a user is authenticated, all operations performed by that user need to be authorized. That is, certain users may be allowed to perform certain tasks, whereas others are not. This is called authorization.

It is recommended that a centralized AAA server be deployed to enforce per-user, AAA-based login authentication for NFVIS access. NFVIS supports RADIUS and TACACS protocols to mediate network access. On the AAA server, only minimum access privileges should be granted to authenticated users according to their specific access requirements. This reduces the exposure to both malicious and unintentional security incidents.

For more information on external authentication, see Configuring RADIUS, on page 21 and Configuring a TACACS+ Server, on page 23.

Role Based Access Control

Limiting network access is important to organizations that have many employees, employ contractors or permit access to third parties, such as customers and vendors. In such a scenario, it is difficult to monitor network access effectively. Instead, it is better to control what is accessible, in order to secure the sensitive data and critical applications.

Role-based access control (RBAC) is a method of restricting network access based on the roles of individual users within an enterprise. RBAC lets users access just the information they need, and prevents them from accessing information that doesn't pertain to them.

An employee's role in the enterprise should be used to determine the permissions granted, in order to ensure that employees with lower privileges can't access sensitive information or perform critical tasks.

The following user roles and privileges are defined in NFVIS

User Role	Privilege
Administrators	Can configure all available features and perform all tasks including changing of user roles. The administrator cannot delete basic infrastructure that is fundamental to NFVIS. The Admin user’s role cannot be changed; it is always "administrators".
Operators	Can Start and stop a VM, and view all information.
Auditors	They are the least privileged users. They have Read-only permission and therefore, can’t modify any configuration.

Benefits of RBAC

There are a number of benefits to using RBAC to restrict unnecessary network access based on people's roles within an organization, including:

- Improving operational efficiency.

  Having predefined roles in RBAC makes it is easy to include new users with the right privileges or switch roles of existing users. It also cuts down on the potential for error when user permissions are being assigned.

- Enhancing compliance.

  Every organization must comply with local, state and federal regulations. Companies generally prefer to implement RBAC systems to meet the regulatory and statutory requirements for confidentiality and privacy because executives and IT departments can more effectively manage how the data is accessed and used. This is particularly important for financial institutions and healthcare companies that manage sensitive data.

- Reducing costs.

  By not allowing user access to certain processes and applications, companies may conserve or use resources such as network bandwidth, memory and storage in a cost-effective manner.

- Decreasing risk of breaches and data leakage.

  Implementing RBAC means restricting access to sensitive information, thus reducing the potential for data breaches or data leakage.

Best practices for role-based access control implementations

- As an administrator, determine the list of users and assign the users to the predefined roles. For example, the user "networkadmin" can be created and added to the user group "administrators".

```
configure terminal
rbac authentication users create-user name networkadmin password Test1_pass role
```

```
administrators
commit
```

Note The user groups or roles are created by the system. You cannot create or modify a user group.

To change the password, use the **rbac authentication users user change-password** command in global configuration mode. To change the user role, use the **rbac authentication users user change-role** command in global configuration mode.

- Terminate accounts for users who no longer require access.

```
configure terminal
rbac authentication users delete-user name test1
```

- Periodically conduct audits to evaluate the roles, the employees who are assigned to them and the access that's permitted for each role. If a user is found to have unnecessary access to a certain system, change the user's role.

For more details see, Users, Roles and Authentication, on page 17

Restrict Device Accessibility

Users have repeatedly been caught unawares by attacks against features they had not protected because they did not know that those features were enabled. Unused services tend to be left with default configurations which are not always secure. These services may also be using default passwords. Some services can give an attacker easy access to information on what the server is running or how the network is setup. The following sections describe how NFVIS avoids such security risks:

Attack vector reduction

Any piece of software can potentially contain security vulnerabilities. More software means more avenues for attack. Even if there are no publicly known vulnerabilities at the time of inclusion, vulnerabilities will probably be discovered or disclosed in the future. To avoid such scenarios, only those software packages which are essential for the NFVIS functionality are installed. This helps to limit software vulnerabilities, reduce resource consumption, and reduce extra work when problems are found with those packages. All third-party software included in NFVIS is registered at a central database in Cisco so that Cisco is able to perform a company level organized response (Legal, Security, etc). Software packages are periodically patched in every release for known Common Vulnerabilities and Exposures (CVEs).

Enabling only essential ports by default

Only those services which are absolutely necessary to setup and manage NFVIS are available by default. This removes the user effort needed to configure firewalls and deny access to unnecessary services. The only services that are enabled by default are listed below along with the ports they open.

Open Port	Service	Description
22/TCP	SSH	Secure Socket Shell for remote command-line access to NFVIS

Open Port	Service	Description
80/TCP	HTTP	Hypertext Transfer Protocol for the NFVIS portal access. All HTTP traffic received by NFVIS is redirected to port 443 for HTTPS
443/TCP	HTTPS	Hypertext Transfer Protocol Secure for secure NFVIS portal access
830/TCP	NETCONF-ssh	Port opened for the Network Configuration Protocol (NETCONF) over SSH. NETCONF is a protocol used for automated configuration of NFVIS and for receiving asynchronous event notifications from NFVIS.
161/UDP	SNMP	Simple Network Management Protocol (SNMP). Used by NFVIS to communicate with remote network-monitoring applications. For more information see, Introduction about SNMP, on page 182

Restrict Access To Authorized Networks For Authorized Services

Only authorized originators should be permitted to even attempt device management access, and access should be only to the services they are authorized to use. NFVIS can be configured such that access is restricted to known, trusted sources and expected management traffic profiles. This reduces the risk of unauthorized access and the exposure to other attacks, such as brute force, dictionary, or DoS attacks.

To protect the NFVIS management interfaces from unnecessary and potentially harmful traffic, an admin user can create Access Control Lists (ACLs) for the network traffic that is received. These ACLs specify the source IP addresses/networks from which the traffic originates, and the type of traffic that is permitted or rejected from these sources. These IP traffic filters are applied to each management interface on NFVIS. The following parameters are configured in an IP receive Access Control List (ip-receive-acl)

Parameter	Value	Description
Source network/Netmask	Network/netmask. For example: 0.0.0.0/0 172.39.162.0/24	This field specifies the IP address/network from which the traffic originates

Parameter	Value	Description
Service	https icmp netconf scpd snmp ssh	Type of traffic from the specified source.
Action	accept drop reject	Action to be taken on the traffic from the source network. With **accept** , new connection attempts will be granted. With **reject** , connection attempts will not be accepted. If the rule is for a TCP based service such as HTTPS, NETCONF, SCP, SSH, the source will get a TCP reset (RST) packet. For non-TCP rules such as SNMP and ICMP, the packet will be dropped. With **drop**, all packets will be dropped immediately, there is no information sent to the source.
Priority	A numeric value	The **priority** is used to enforce an order on the rules. Rules with a higher numeric value for priority will be added further down in the chain. If you want to make sure that a rule will be added after another one, use a low priority number for the first and a higher priority number for the following.

The following sample configurations illustrate some scenarios that can be adapted for specific use-cases.

Configuring the IP Receive ACL

The more restrictive an ACL, the more limited the exposure to unauthorized access attempts. However, a more restrictive ACL can create a management overhead, and can impact accessibility to perform troubleshooting. Consequently, there is a balance to be considered. One compromise is to restrict access to internal corporate IP addresses only. Each customer must evaluate the implementation of ACLs in relation to their own security policy, risks, exposure, and acceptance thereof.

Reject ssh traffic from a subnet:

```
nfvis(config)# system settings ip-receive-acl 171.70.63.0/24 service ssh action reject
priority 1
```

Removing ACLs:

When an entry is deleted from **ip-receive-acl**, all configurations to that source are deleted since the source IP address is the key. To delete just one service, configure other services again.

```
nfvis(config)# no system settings ip-receive-acl 171.70.63.0/24
```

For more details see, Configuring the IP Receive ACL, on page 12

Privileged Debug Access

The super-user account on NFVIS is disabled by default, to prevent all unrestricted, potentially adverse, system-wide changes and NFVIS does not expose the system shell to the user.

However, for some hard to debug issues on the NFVIS system, the Cisco Technical Assistance Center team (TAC) or development team might require shell access to the customer's NFVIS. NFVIS has a secure unlock infrastructure to ensure that privileged debug access to a device in the field is restricted to authorized Cisco employees. To securely access the Linux shell for this kind of interactive debugging, a challenge-response authentication mechanism is used between NFVIS and the Interactive debugging server maintained by Cisco. The admin user's password is also required in addition to the challenge-response entry to ensure that the device is accessed with the customer's consent.

Steps to access the shell for Interactive Debugging:

1. An admin user initiates this procedure using this hidden command.

   ```
   nfvis# system shell-access
   ```

2. The screen will show a challenge string, for example:

   ```
   Challenge String (Please copy everything between the asterisk lines exclusively):

   ********************************************************************************
   SPH//wkAAABORlZJU0VOQ1M1NDA4L0s5AQAAABt+dcx+hB0V06r9RkdMMjEzNTgw
   RlHq7BxeAAA=
   DONE.
   ********************************************************************************
   ```

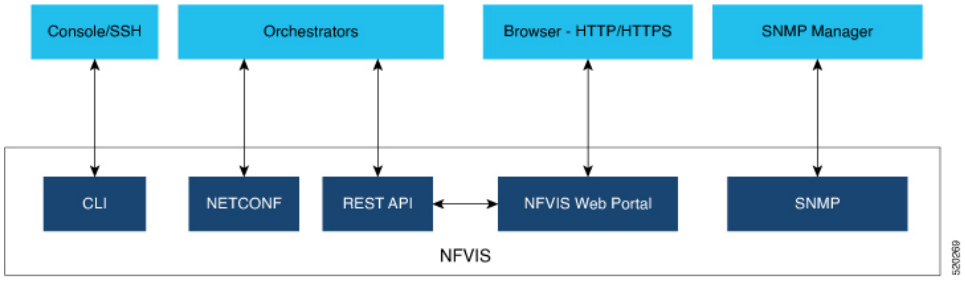
3. The Cisco member enters the Challenge string on an Interactive Debug server maintained by Cisco. This server verifies that the Cisco user is authorized to debug NFVIS using the shell, and then returns a response string.
4. Enter the response string on the screen below this prompt:

   Input your response when ready:

5. When prompted, the customer should enter the admin password.
6. You get shell-access if the password is valid.
7. Development or TAC team uses the shell to proceed with the debugging.
8. To exit shell-access type **Exit**.

Secure Interfaces

NFVIS management access is allowed using the interfaces shown in the diagram. The following sections describe security best practices for these interfaces to NFVIS.



Console

The console port is an asynchronous serial port that allows you to connect to the NFVIS CLI for initial configuration. A user can access the console with either physical access to the NFVIS or remote access through the use of a terminal server. If console port access is required via a terminal server, configure access lists on the terminal server to allow access only from the required source addresses.

SSH

Users can access the NFVIS CLI by using SSH as a secure means of remote login. The integrity and confidentiality of NFVIS management traffic is essential to the security of the administered network since administration protocols frequently carry information which could be used to penetrate or disrupt the network.

NFVIS uses SSH version 2, which is Cisco's and the Internet's de facto standard protocol for interactive logins and supports strong encryption, hash, and key exchange algorithms recommended by the Security and Trust Organization within Cisco.

CLI Session timeout

By logging in via SSH, a user establishes a session with NFVIS. While the user is logged in, if the user leaves the logged-in session unattended, this can expose the network to a security risk. Session security limits the risk of internal attacks, such as one user trying to use another user's session.

To mitigate this risk, NFVIS times out CLI sessions after 15 minutes of inactivity. When the session timeout is reached, the user is automatically logged out.

NETCONF

The Network Configuration Protocol (NETCONF) is a Network Management protocol developed and standardized by the IETF for the automated configuration of network devices.

The NETCONF protocol uses an Extensible Markup Language (XML) based data encoding for the configuration data as well as the protocol messages. The protocol messages are exchanged on top of a secure transport protocol.

NETCONF allows NFVIS to expose an XML-based API that the network operator can use to set and get configuration data and event notifications securely over SSH.

For more information see, NETCONF Event Notifications, on page 181.

REST API

NFVIS can be configured using RESTful API over HTTPS. The REST API allow the requesting systems to access and manipulate the NFVIS configuration by using a uniform and predefined set of stateless operations. Details on all the REST APIs can be found in the NFVIS API Reference guide.

When the user issues a REST API, a session is established with NFVIS. In order to limit risks related to denial of service attacks, NFVIS limits the total number of concurrent REST sessions to 100.

NFVIS Web Portal

The NFVIS portal is a web-based Graphical User Interface that displays information about NFVIS. The portal presents the user with an easy means to configure and monitor NFVIS over HTTPS without having to know the NFVIS CLI and API.

Session Management

The stateless nature of HTTP and HTTPS requires a method of uniquely tracking users through the use of unique session IDs and cookies.

NFVIS encrypts the user's session. The AES-256-CBC cipher is used to encrypt the session contents with an HMAC-SHA-256 authentication tag. A random 128-bit Initialization Vector is generated for each encryption operation.

An Audit record is started when a portal session is created. Session information is deleted when the user logs out or when the session times out.

The default idle timeout for portal sessions is 15 minutes. However, this can be configured for the current session to a value between 5 and 60 minutes on the Settings page. Auto-logout will be initiated after this period. Multiple sessions are not permitted in a single browser. The Maximum number of concurrent sessions are set to 30.

The NFVIS portal utilizes cookies to associate data with the user. It uses the following cookie properties for enhanced security:

- **ephemeral** to ensure the cookie expires when the browser is closed
- **httpOnly** to make the cookie inaccessible from JavaScript
- **secureProxy** to ensure the cookie can only be sent over SSL.

Even after authentication, attacks such as Cross-Site Request Forgery (CSRF) are possible. In this scenario, an end user might inadvertently execute unwanted actions on a web application in which they're currently authenticated. To prevent this, NFVIS uses **CSRF** tokens to validate every REST API that is invoked during each session.

URL Redirection

In typical web servers, when a page is not found on the web server, the user gets a 404 message; for pages that exist, they get a login page. The security impact of this is that an attacker can perform a brute force scan and easily detect which pages and folders exist.

To prevent this on NFVIS, all non-existent URLs prefixed with the device IP are redirected to the portal login page with a 301 status response code. This means that irrespective of the URL requested by an attacker, they will always get the login page to authenticate themselves.

All HTTP server requests are redirected to HTTPS and have the following headers configured:

- X-Content-Type-Options
- X-XSS-Protection
- Content-Security-Policy
- X-Frame-Options
- Strict-Transport-Security
- Cache-Control

Disabling the Portal

The NFVIS portal access is enabled by default. If you are not planning to use the portal, it is recommended to disable portal access using this command:

```
Configure terminal
System portal access disabled
commit
```

HTTPS

All the HTTPS data to and from NFVIS uses Transport Layer Security (TLS) to communicate across the network. TLS is the successor to Secure Socket Layer (SSL).

The TLS handshake involves authentication during which the client verifies the server's SSL certificate with the certificate authority that issued it. This confirms that the server is who it says it is, and that the client is interacting with the owner of the domain.

By default, NFVIS uses a self-signed certificate to prove its identity to its clients. This certificate has a 2048-bit public key to increase the security of the TLS encryption, since the encryption strength is directly related to the key size.

Certificate Management

NFVIS generates a self-signed SSL certificate when first installed. It is a security best practice to replace this certificate with a valid certificate signed by a compliant Certificate Authority (CA).

Use the following steps to replace the default self-signed certificate:

1. Generate a Certificate Signing Request (CSR) on NFVIS.

   A Certificate Signing request (CSR) is a file with a block of encoded text that is given to a Certificate Authority when applying for an SSL Certificate. This file contains information that should be included in the certificate such as the organization name, common name (domain name), locality, and country. The file also contains the public key that should be included in the certificate. NFVIS uses a 2048-bit public key since encryption strength is higher with a higher key size.

   To generate a CSR on NFVIS, run the following command:

   ```
   nfvis# system certificate signing-request [common-name  country-code  locality
   organization  organization-unit-name  state]
   ```

   The CSR file is saved as */data/intdatastore/download/nfvis.csr*.

   .
2. Get an SSL certificate from a CA using the CSR.

From an external host, use the scp command to download the Certificate Signing Request.

```
[myhost:/tmp] > scp -P 22222 admin@<NFVIS-IP>:/data/intdatastore/download/nfvis.csr
<destination-file-name>
```

Contact a Certificate authority to issue a new SSL server certificate using this CSR.

3. Install the CA Signed Certificate.

   From an external server, use the scp command to upload the certificate file into NFVIS to the *data/intdatastore/uploads/* directory.

   ```
   [myhost:/tmp] > scp -P 22222 <certificate file>
   admin@<NFVIS-IP>:/data/intdatastore/uploads
   ```

   Install the certificate in NFVIS using the following command.

   ```
   nfvis# system certificate install-cert path file:///data/intdatastore/uploads/<certificate
    file>
   ```

4. Switch to using the CA Signed Certificate.

   Use the following command to start using the CA signed certificate instead of the default self-signed certificate.

   ```
   nfvis(config)# system certificate use-cert cert-type ca-signed
   ```

SNMP Access

Simple Network Management Protocol (SNMP) is an Internet Standard protocol for collecting and organizing information about managed devices on IP networks, and for modifying that information to change device behavior.

Three significant versions of SNMP have been developed. NFVIS supports SNMP version 1, version 2c and version 3. SNMP versions 1 and 2 use community strings for authentication, and these are sent in plain-text. So, it is a security best practice to use SNMP v3 instead.

SNMPv3 provides secure access to devices by using three aspects: - users, authentication, and encryption. SNMPv3 uses the USM (User-based Security Module) for controlling access to information available via SNMP. The SNMP v3 user is configured with an authentication type, a privacy type as well as a passphrase. All users sharing a group utilize the same SNMP version, however, the specific security level settings (password, encryption type, etc.) are specified per-user.

The following table summarizes the security options within SNMP

Model	Level	Authentication	Encyption	Outcome
v1	noAuthNoPriv	Community String	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community String	No	Uses a community string match for authentication.

Model	Level	Authentication	Encyption	Outcome
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the HMAC-MD5-96 or HMAC-SHA-96 algorithms.
v3	authPriv	MD5 or SHA	Data Encryption Standard (DES) or Advanced Encryption Standard (AES)	Provides authentication based on the HMAC-MD5-96 or HMAC-SHA-96 algorithms. Provides DES Cipher algorithm in Cipher Block Chaining Mode (CBC-DES) or AES encryption algorithm used in Cipher FeedBack Mode (CFB), with a 128-bit key size(CFB128-AES-128)

Since its adoption by NIST, AES has become the dominant encryption algorithm throughout the industry. To follow the industry's migration away from MD5 and toward SHA, it is a security best practice to configure the SNMP v3 authentication protocol as SHA and privacy protocol as AES.

For more details on SNMP see, Introduction about SNMP, on page 182

Legal Notification Banners

It is recommended that a legal notification banner is present on all interactive sessions to ensure that users are notified of the security policy being enforced and to which they are subject. In some jurisdictions, civil and/or criminal prosecution of an attacker who breaks into a system is easier, or even required, if a legal notification banner is presented, informing unauthorized users that their use is in fact unauthorized. In some jurisdictions, it may also be forbidden to monitor the activity of an unauthorized user unless they have been notified of the intent to do so.

Legal notification requirements are complex and vary in each jurisdiction and situation. Even within jurisdictions, legal opinions vary. Discuss this issue with your own legal counsel to ensure that the notification banner meets company, local, and international legal requirements. This is often critical to securing appropriate

action in the event of a security breach. In cooperation with the company legal counsel, statements which may be included in a legal notification banner include:

- Notification that the system access and use is permitted only by specifically authorized personnel, and perhaps information about who may authorize use.
- Notification that unauthorized access and use of the system is unlawful, and may be subject to civil and/or criminal penalties.
- Notification that access and use of the system may be logged or monitored without further notice, and the resulting logs may be used as evidence in court.
- Additional specific notices required by specific local laws.

From a security rather than a legal point of view, a legal notification banner should not contain any specific information about the device, such as its name, model, software, location, operator or owner because this kind of information may be useful to an attacker.

The following is a sample legal notification banner which can be displayed before login:

```
UNAUTHORIZED ACCESS TO THIS DEVICE IS PROHIBITED You must have explicit, authorized permission
 to access or configure this device. Unauthorized attempts and actions to access or use
this system may result in civil and/or criminal penalties. All activities performed on this
 device are logged and monitored
```

Note Present a legal notification banner approved by company legal counsel.

NFVIS allows the configuration of a banner and Message of the Day (MOTD). The banner is displayed before the user logs in. Once the user logs in to NFVIS, a system-defined banner provides Copyright information about NFVIS, and the message-of-the-day (MOTD), if configured, will appear, followed by the command line prompt or portal view, depending on the login method.

It is recommended that a login banner is implemented to ensure that a legal notification banner is presented on all the device management access sessions prior to a login prompt being presented. Use this command to configure the banner and MOTD.

```
nfvis(config)# banner-motd banner <banner-text> motd <message-of-the-day-text>
```

For more information about the banner command, see Configure Banner, Message of the day and System Time, on page 59.

Factory Default Reset

Factory Reset removes all the customer specific data that has been added to the device since the time of its shipping. The data erased includes configurations, log files, VM images, connectivity information, and user login credentials.

It provides one command to reset the device to factory-original settings, and is useful in the following scenarios:

- Return Material Authorization (RMA) for a device—If you have to return a device to Cisco for RMA, use Factory Default reset to remove all the customer-specific data.
- Recovering a compromised device— If the key material or credentials stored on a device is compromised, reset the device to factory configuration and then reconfigure the device.

- If the same device needs to be re-used at a different site with a new configuration, perform a Factory Default reset to remove the existing configuration and bring it to a clean state.

NFVIS provides the following options within Factory default reset:

Factory Reset Option	Data Erased	Data Retained
all	All configuration, uploaded image files, VMs and logs. Connectivity to the device will be lost.	The admin account is retained and the password will be changed to the factory default password.
all-except-images	All configuration except image configuration, VMs, and uploaded image files. Connectivity to the device will be lost.	Image configuration, registered images and logs The admin account is retained and the password will be changed to the factory default password.
all-except-images-connectivity	All configuration except image, network and connectivity configuration, VMs, and uploaded image files. Connectivity to the device is available.	Images, network and connectivity related configuration, registered images, and logs. The admin account is retained and the previously configured admin password will be preserved.
manufacturing	All configuration except image configuration, VMs, uploaded image files, and logs. Connectivity to the device will be lost.	Image related configuration and registered images The admin account is retained and the password will be changed to the factory default password.

The user must choose the appropriate option carefully based on the purpose of the Factory Default reset.

For more information, see Resetting to Factory Default, on page 58.

Infrastructure Management Network

An infrastructure management network refers to the network carrying the control and management plane traffic (such as NTP, SSH, SNMP, syslog, etc.) for the infrastructure devices. Device access can be through the console, as well as through the Ethernet interfaces. This control and management plane traffic is critical to network operations, providing visibility into and control over the network. Consequently, a well-designed and secure infrastructure management network is critical to the overall security and operations of a network. One of the key recommendations for a secure infrastructure management network is the separation of management and data traffic in order to ensure remote manageability even under high load and high traffic conditions. This can be achieved using a dedicated management interface.

The following are the Infrastructure management network implementation approaches:

Out-of-band Management

An Out-of-band Management (OOB) management network consists of a network which is completely independent and physically disparate from the data network that it helps to manage. This is also sometimes referred to as a Data Communications Network (DCN). Network devices can connect to the OOB network in different ways: – NFVIS supports a built-in management interface that can be used to connect to the OOB network. NFVIS allows the configuration of a predefined physical interface, the MGMT port on the ENCS, as a dedicated management interface. Restricting management packets to designated interfaces provides greater control over the management of a device, thereby providing more security for that device. Other benefits include improved performance for data packets on non-management interfaces, support for network scalability, need for fewer access control lists (ACLs) to restrict access to a device, and prevention of management packet floods from reaching the CPU.

Network devices can also connect to the OOB network via dedicated data interfaces. In this case, ACLs should be deployed to ensure that management traffic is only handled by the dedicated interfaces.

For further information, see Configuring the IP Receive ACL, on page 12 and Port 22222 and Management Interface ACL, on page 12.

Pseudo out-of-band Management

A pseudo out-of-band management network uses the same physical infrastructure as the data network but provides logical separation through the virtual separation of traffic, by using VLANs. NFVIS supports creating VLANs and virtual bridges to help identify different sources of traffic and separate traffic between VMs. Having separate bridges and VLANs isolates the virtual machine network's data traffic and the management network, thus providing traffic segmentation between the VMs and the host. For further information see Configuring VLAN for NFVIS Management Traffic, on page 11.

In-band Management

An in-band management network uses the same physical and logical paths as the data traffic.

Ultimately, this network design requires a per-customer analysis of risk versus benefits and costs. Some general considerations include:

- An isolated OOB management network maximizes visibility and control over the network even during disruptive events.
- Transmitting network telemetry over an OOB network minimizes the chance for disruption of the very information which provides critical network visibility.
- In-band management access to network infrastructure, hosts, etc. is vulnerable to complete loss in the event of a network incident, removing all the network visibility and control. Appropriate QoS controls should be put in place to mitigate this occurrence.
- NFVIS features interfaces which are dedicated to device management, including serial console ports and Ethernet management interfaces.
- An OOB management network can typically be deployed at a reasonable cost, since management network traffic does not typically demand high bandwidth nor high performance devices, and only requires sufficient port density to support the connectivity to each infrastructure device.

Locally Stored Information Protection

Protecting Sensitive Information

NFVIS stores some sensitive information locally, including passwords and secrets. Passwords should generally be maintained and controlled by a centralized AAA server. However, even if a centralized AAA server is deployed, some locally-stored passwords are required for certain cases such as local fallback in the case of AAA servers not being available, special-use usernames, etc. These local passwords and other sensitive information are stored on NFVIS as hashes so that it is not possible to recover the original credentials from the system. Hashing is a widely accepted industry norm.

File Transfer

Files which may need to be transferred to NFVIS devices include VM image and NFVIS upgrade files. The secure transfer of files is critical for network infrastructure security. NFVIS supports Secure Copy (SCP) to ensure the security of file transfer. SCP relies on SSH for secure authentication and transport, enabling the secure and authenticated copying of files.

A secure copy from NFVIS is initiated through the scp command. The secure copy (scp) command allows only the admin user to securely copy files from NFVIS to an external system, or from an external system to NFVIS.

The syntax for the scp command is:

```
scp <source> <destination>
```

We use port 22222 for the NFVIS SCP server. By default, this port is closed and users cannot secure copy files into NFVIS from an external client. If there is a need to SCP a file from an external client, the user can open the port using:

```
system settings ip-receive-acl (address)/(mask lenth)  service scpd priority (number) action
 accept
commit
```

To prevent users from accessing system directories, secure copy can be performed only to or from intdatastore:, extdatastore1:, extdatastore2:, usb: and nfs:, if available. Secure copy can also be performed from logs: and techsupport:

Logging

NFVIS access and configuration changes are logged as audit logs to record the following information:

- Who accessed the device
- When did a user log in
- What did a user do in terms of the host configuration and the VM lifecycle
- When did a user log off

- Failed access attempts
- Failed authentication requests
- Failed authorization requests

This information is invaluable for forensic analysis in case of unauthorized attempts or access, as well as for configuration change issues and to help plan group administration changes. It may also be used real time to identify anomalous activities which may indicate that an attack is taking place. This analysis can be correlated with information from additional external sources, such as IDS and firewall logs.

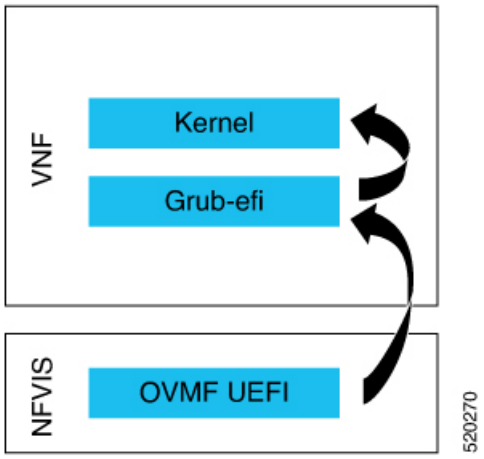
All the key events on the NFVIS are sent as event notifications to NETCONF subscribers and as syslogs to the configured central logging servers. For more information on syslog messages and event notifications, see Appendix, on page 207.

Virtual Machine security

This section describes security features related to the registration, deployment and operation of Virtual Machines on NFVIS.

VNF secure boot

NFVIS supports Open Virtual Machine Firmware (OVMF) to enable UEFI secure boot for Virtual Machines which support secure boot. VNF Secure boot verifies that each layer of the VM boot software is signed, including the bootloader, the operating system kernel, and operating system drivers.



For more information see, Secure Boot of VNFs, on page 115.

VNC Console Access Protection

NFVIS allows the user to create a Virtual Network Computing (VNC) session to access a deployed VM's remote desktop. To enable this, NFVIS dynamically opens a port to which the user can connect using their web browser. This port is only left open for 60 seconds for an external server to start a session to the VM. If no activity is seen within this time, the port is closed. The port number is assigned dynamically and thereby allows only a one-time access to the VNC console.

```
nfvis# vncconsole start deployment-name 1510614035 vm-name ROUTER
vncconsole-url :6005/vnc_auto.html
```

Pointing your browser to https://<nfvis ip>:6005/vnc_auto.html will connect to the ROUTER VM's VNC console.

Encrypted VM config data variables

During VM deployment, the user provides a day-0 configuration file for the VM. This file can contain sensitive information such as passwords and keys. If this information is passed as clear text, it appears in log files and internal database records in clear text. This feature allows the user to flag a config data variable as sensitive so that its value is encrypted using AES-CFB-128 encryption before it is stored or passed to internal subsystems.

For more information see, VM Deployment Parameters, on page 100.

Checksum verification for Remote Image Registration

To register a remotely located VNF image, the user specifies its location. The image will need to be downloaded from an external source, such as an NFS server or a remote HTTPS server.

To know if a downloaded file is safe to install, it is essential to compare the file's checksum before using it. Verifying the checksum helps ensure that the file was not corrupted during network transmission, or modified by a malicious third party before you downloaded it.

NFVIS supports the checksum and checksum_algorithm options for the user to provide the expected checksum and checksum algorithm (SHA256 or SHA512) to be used to verify the checksum of the downloaded image. Image creation fails if the checksum does not match.

Certification Validation for Remote Image Registration

To register a VNF image located on a HTTPS server, the image will need to be downloaded from the remote HTTPS server. To securely download this image, NFVIS verifies the SSL certificate of the server. The user needs to specifying either the path to the certificate file or the PEM format certificate contents to enable this secure download.

More details can be found at Register a Remote VM Image

VM Isolation and Resource provisioning

The Network Function Virtualization (NFV) architecture consists of:

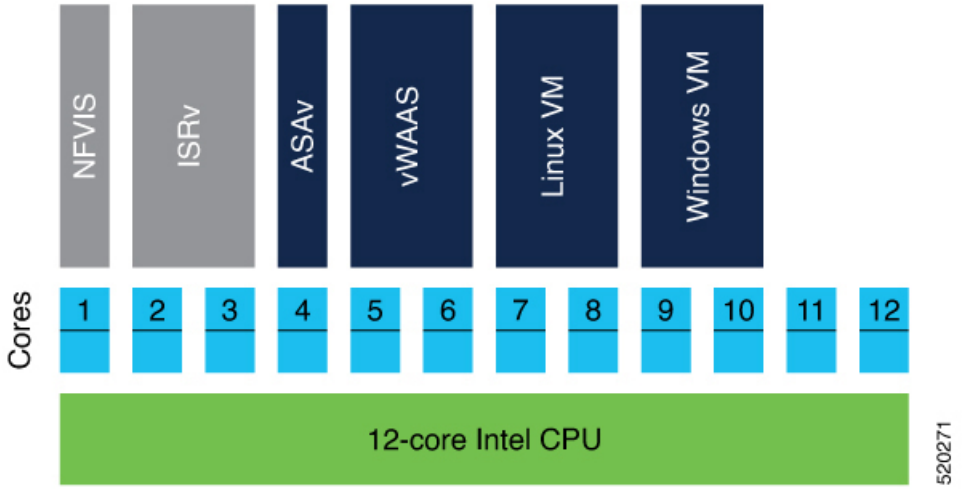
- Virtualized network functions (VNFs), which are Virtual Machines running software applications that deliver network functionality such as a router, firewall, load balancer, and so on.
- Network functions virtualization infrastructure, which consists of the infrastructure components—compute, memory, storage, and networking, on a platform that supports the required software and hypervisor.

With NFV, network functions are virtualized so that multiple functions can be run on a single server. As a result, less physical hardware is needed, allowing for resource consolidation. In this environment, it is essential to simulate dedicated resources for multiple VNFs from a single, physical hardware system. Using NFVIS, VMs can be deployed in a controlled manner such that each VM receives the resources it needs. Resources

are partitioned as needed from the physical environment to the many virtual environments. The individual VM domains are isolated so they are separate, distinct, and secure environments, which are not contending with each other for shared resources.

VMs cannot use more resources than provisioned. This avoids a Denial of Service condition from one VM consuming the resources. As a result, CPU, memory, network and storage are protected.

CPU Isolation



The NFVIS system reserves cores for the infrastructure software running on the host. The rest of the cores are available for VM deployment. This guarantees that the VM's performance does not affect the NFVIS host performance.

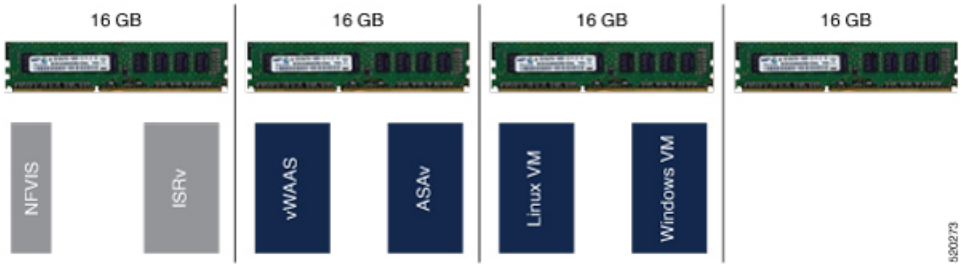
Low-latency VMs

NFVIS explicitly assigns dedicated cores to low latency VMs that are deployed on it. If the VM requires 2 vCPUs, it is assigned 2 dedicated cores. This prevents sharing and oversubscription of cores and guarantees the performance of the low-latency VMs. If the number of available cores is less than the number of vCPUs requested by another low-latency VM, the deployment is prevented since we do not have sufficient resources.

Non low-latency VMs

NFVIS assigns sharable CPUs to non low latency VMs. If the VM requires 2 vCPUs, it is assigned 2 CPUs. These 2 CPUs are shareable among other non low latency VMs. If the number of available CPUs is less than the number of vCPUs requested by another non low-latency VM, the deployment is still allowed because this VM will share the CPU with existing non low latency VMs.

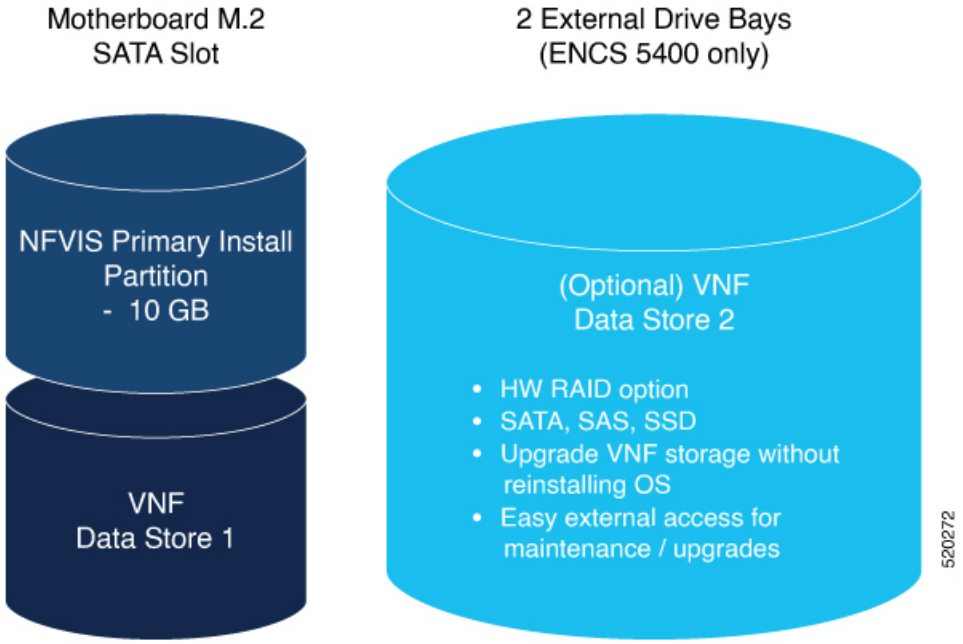
Memory Allocation



The NFVIS Infrastructure requires a certain amount of memory. When a VM is deployed, there is a check to ensure that the memory available after reserving the memory required for the infrastructure and previously deployed VMs, is sufficient for the new VM. We do not allow memory oversubscription for the VMs.

VMs are not allowed to directly access the host file system and storage.

Storage Isolation



The ENCS platform supports an internal datastore (M2 SSD) and external disks. NFVIS is installed on the internal datastore. VNFs can also be deployed on this internal datastore. It is a security best practice to store customer data and deploy customer application Virtual Machines on the external disks. Having physically separate disks for the system files vs the application files helps to protect system data from corruption and security issues.

-

Interface Isolation

Single Root I/O Virtualization or SR-IOV is a specification that allows the isolation of PCI Express (PCIe) resources such as an Ethernet port. Using SR-IOV a single Ethernet port can be made to appear as multiple, separate, physical devices known as Virtual Functions. All of the VF devices on that adapter share the same physical network port. A guest can use one or more of these Virtual Functions. A Virtual Function appears to the guest as a network card, in the same way as a normal network card would appear to an operating system.

Virtual Functions have near-native performance and provide better performance than para-virtualized drivers and emulated access. Virtual Functions provide data protection between guests on the same physical server as the data is managed and controlled by the hardware.

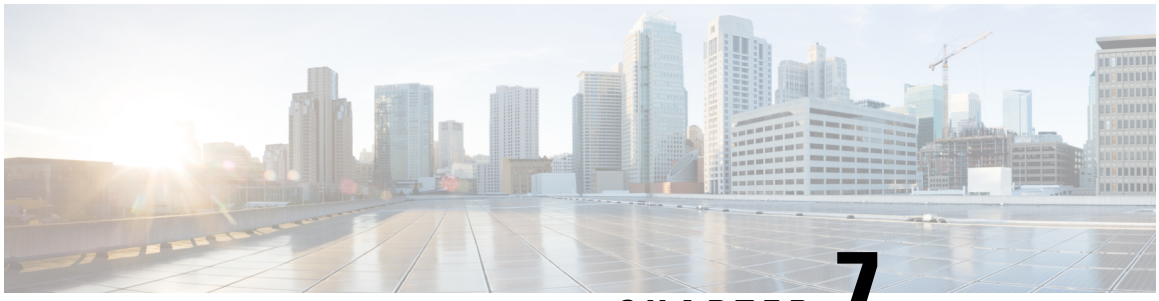
NFVIS VNFs can use SR-IOV networks to connect to WAN and LAN Backplane ports.

Each such VM owns a virtual interface and its related resources achieving data protection among VMs.

Secure Development Lifecycle

NFVIS follows a Secure Development Lifecycle (SDL) for software. This is a repeatable, measurable process designed to reduce vulnerabilities and enhance the security and resilience of Cisco solutions. Cisco SDL applies industry-leading practices and technology to build trustworthy solutions that have fewer field-discovered product security incidents. Every NFVIS release goes through the following processes.

- Following Cisco-internal and market-based Product Security Requirements
- Registering 3rd party software with a central repository at Cisco for vulnerability tracking
- Periodically patching software with known fixes for CVEs.
- Designing software with Security in mind
- Following secure coding practices such as using vetted common security modules like CiscoSSL, running Static Analysis and implementing input validation for Preventing command injection, etc.
- Using Application Security tools such as IBM AppScan, Nessus, and other Cisco internal tools.



CHAPTER 7

Platform Specific Configurations

ENCS Switch Configuration

Access to the ENCS switch is restricted through Consent Token. Consent Token is a security feature that is used to authenticate the network administrator of an organization to access system shell with mutual consent from the network administrator and Cisco Technical Assistance Centre (Cisco TAC).

Note From the switch console, there is access to debug mode and an advanced debug mode. Credentials of the local user are synchronized to access debug mode. Advanced debug uses unique credentials for each device that allows for additional debugging options for Cisco engineering. To enter either debug mode permission must be granted through Consent Token.

ENCS Switch Commands

See, Cisco Enterprise Network Compute System Switch Command Reference for switch commands.

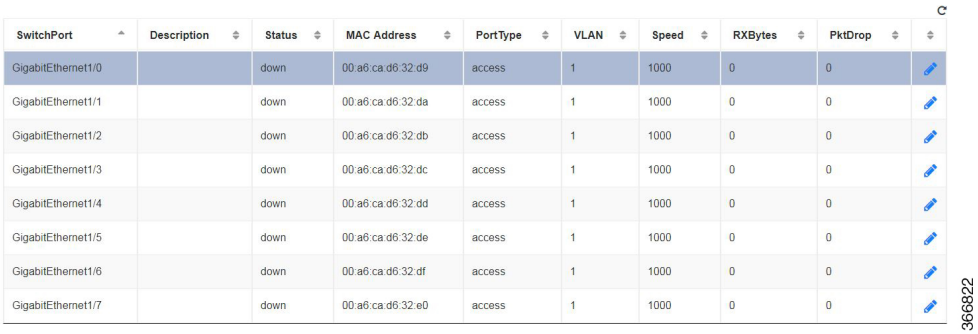
ENCS Switch APIs

See, API Reference for Cisco Enterprise Network Function Virtualization Infrastructure Software for switch related APIs.

ENCS Switch Portal Configuration

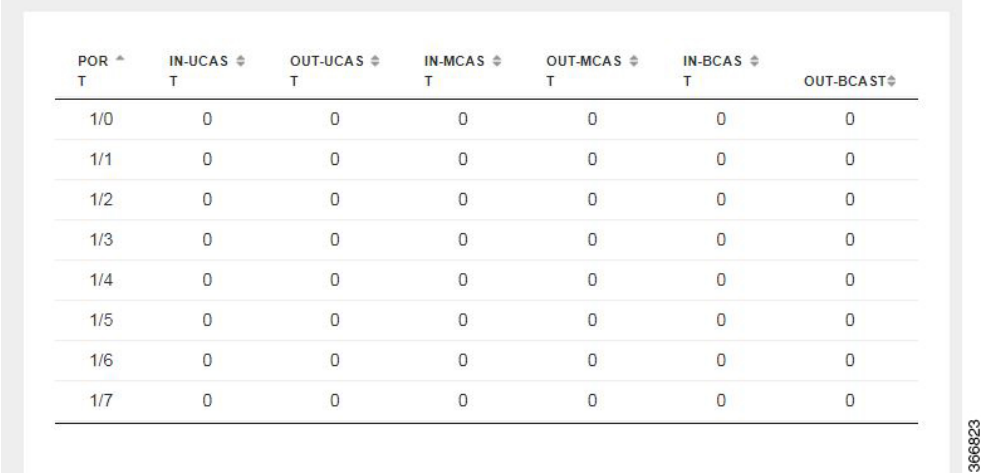
Switch Settings

The **Switch** option from the Cisco Enterprise NFVIS portal allows you to configure STP/RSTP, VLAN on specified ranges, RADIUS based authentication, and port channel load balancing for various switch ports. This section describes how to configure settings on the ENCS switch portal.



SwitchPort	Description	Status	MAC Address	PortType	VLAN	Speed	RXBytes	PktDrop	
GigabitEthernet1/0		down	00:a6:ca:d6:32:d9	access	1	1000	0	0	
GigabitEthernet1/1		down	00:a6:ca:d6:32:da	access	1	1000	0	0	
GigabitEthernet1/2		down	00:a6:ca:d6:32:db	access	1	1000	0	0	
GigabitEthernet1/3		down	00:a6:ca:d6:32:dc	access	1	1000	0	0	
GigabitEthernet1/4		down	00:a6:ca:d6:32:dd	access	1	1000	0	0	
GigabitEthernet1/5		down	00:a6:ca:d6:32:de	access	1	1000	0	0	
GigabitEthernet1/6		down	00:a6:ca:d6:32:df	access	1	1000	0	0	
GigabitEthernet1/7		down	00:a6:ca:d6:32:e0	access	1	1000	0	0	

366822



PORT	IN-UCAST	OUT-UCAST	IN-MCAST	OUT-MCAST	IN-BCAST	OUT-BCAST
1/0	0	0	0	0	0	0
1/1	0	0	0	0	0	0
1/2	0	0	0	0	0	0
1/3	0	0	0	0	0	0
1/4	0	0	0	0	0	0
1/5	0	0	0	0	0	0
1/6	0	0	0	0	0	0
1/7	0	0	0	0	0	0

366823

You can view the Switch Interface operational data and the statistics parameters in the following table:

Table 9: Switch Settings Interface

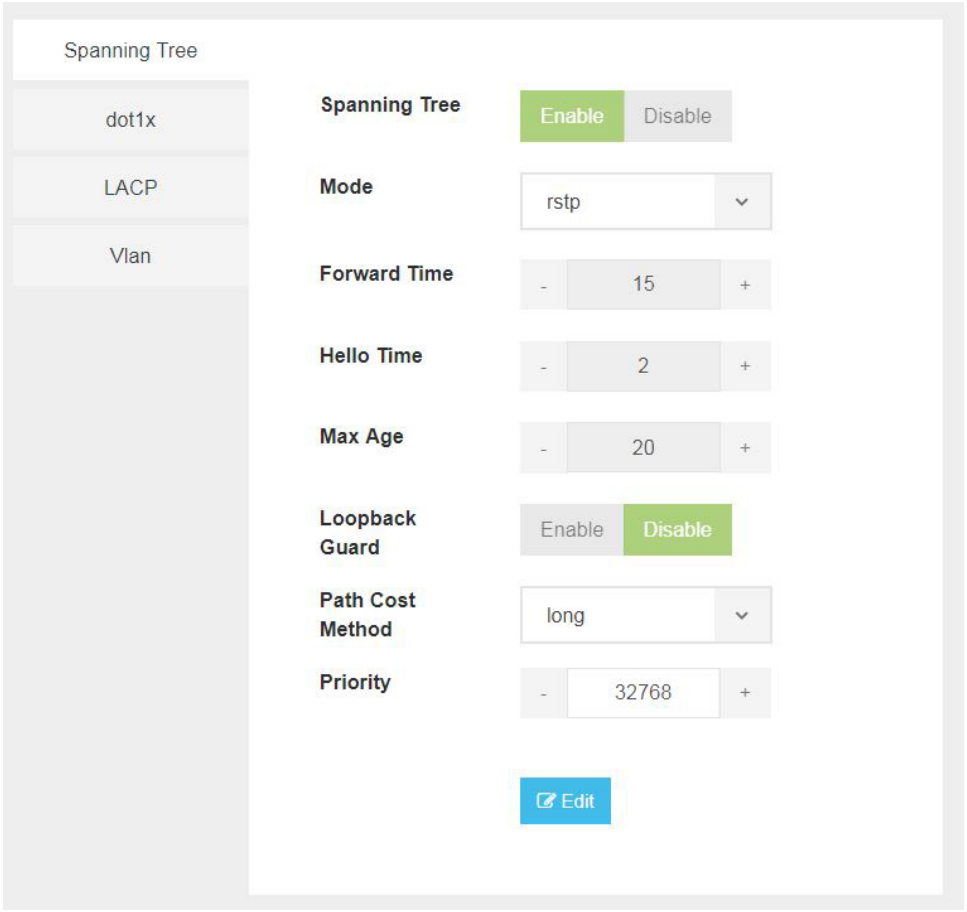
Parameter	Description	Values
SwitchPort	Specifies the switch interface name.	
Description	Specifies the description of the interface.	
Status	Specifies the status of the interface.	up or down
MAC Address	Specifies the MAC address of the interface.	
PortType	Specifies the mode of the port interface.	Supported types are: • access • dot1q-tunnel • private-vlan • trunk
VLAN	Specifies the VLAN ID.	Range: 1-2349 and 2450-4093

Speed	Specifies the speed of the interface.	Speed: • 10 MBPS • 100 MBPS • 1000 MBPS
RxBytes	Specifies the received data on interface in bytes.	
PktDrop	Specifies the number of packet drops.	
PORT	Specifies the port number.	
IN-UCAST	Specifies the number of incoming unicast packets at the interface.	
OUT-UCAST	Specifies the number of outgoing unicast packets at the interface.	
IN-MCAST	Specifies the number of incoming multicast packets at the interface.	
OUT-MCAST	Specifies the number of outgoing multicast packets at the interface.	
IN-BCAST	Specifies the number of incoming broadcast packets at the interface.	
OUT-BCAST	Specifies the number of outgoing broadcast packets at the interface.	

Configuring Spanning Tree

Spanning Tree Protocol (STP) is a Layer 2 protocol that runs on bridges and switches. The main purpose of STP is to ensure that you do not create loops when you have redundant paths in your network.

The Spanning Tree option is enabled by default. You can click on **edit** and make the necessary settings or disable Spanning Tree if required.



The configuration of spanning tree has the following parameters when it is enabled:

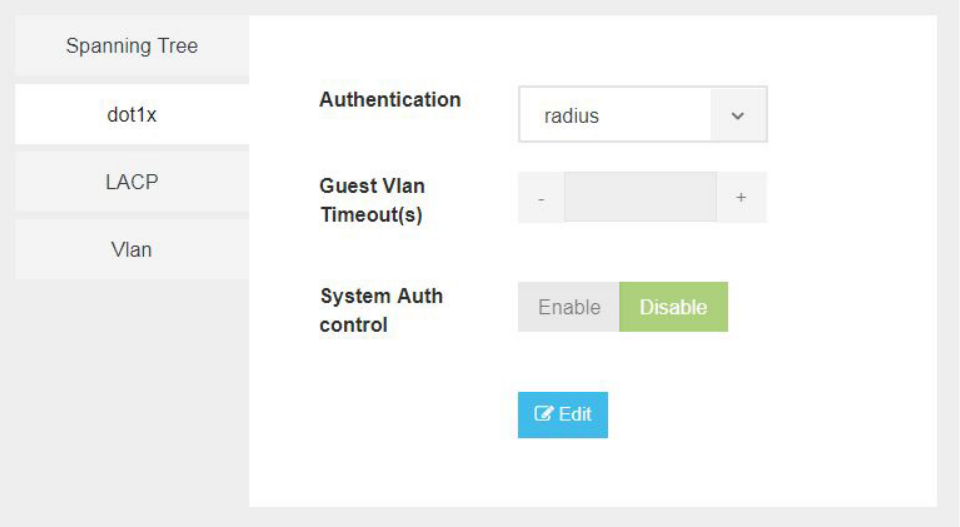
Table 10: Spanning Tree Parameters

Parameter	Description	Values
Spanning Tree	Specifies the state of the Spanning Tree.	Enable or Disable The default value is Enable.
Mode	Specifies the mode of the Spanning Tree.	stp or rstp
Forward Time	Specifies the Spanning Tree forward time in seconds.	Range: 4-30 seconds
Hello Time	Specifies the Hello time in seconds.	Range: 1 to10 seconds
Max Age	Specifies the spanning-tree bridge maximum age in seconds.	Range: 6 to 40 seconds
Loopback Guard	Specifies the loopback guard status.	Enable or Disable

Path Cost Method	Specifies the speed of the interface.	Method: • long - for 32 bit based values for default port path costs. • short - 16 bit based values for default port path costs. The default method is long.
Priority	Specifies the port priority.	Range: 0 to 61440 in steps of 4096 The default value is 32768.
BPDU Filtering	Specifies that BPDU packets are filtered when the spanning tree is disabled on an interface.	
BPDU Flooding	Specifies that BPDU packets are flooded unconditionally when the spanning tree is disabled on an interface.	

Configuring Dot1x

This chapter describes how to configure dot1x port-based authentication on the Cisco Enterprise NFVIS portal. dot1x prevents unauthorized devices (clients) from gaining access to the network. It is a standard for media-level (Layer 2) access control, offering the capability to permit or deny network connectivity based on the identity of the end user or device. The dot1x is disabled by default. You can click on **edit** to enable dot1x.



The configuration of dot1x has the following parameters:

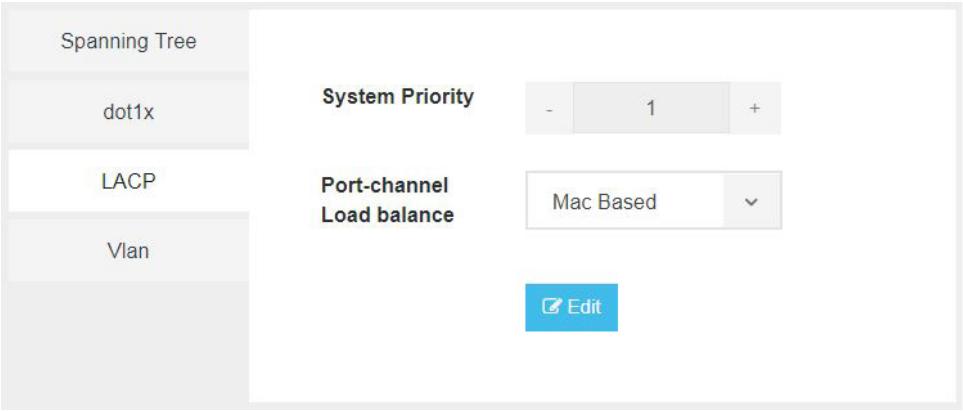
Table 11: Dot1x Parameters

Parameter	Description	Values

Authentication	Specifies the authentication type for the port.	radius or none The default value is radius.
Guest VLAN Timeout(s)	Specifies the time delay in seconds between enabling Dot1X (or port up) and adding the port to the guest VLAN.	Range: 30 to 180 seconds
System Auth control	Specifies the authentication control.	Enable or Disable

Configuring LACP

The Link Aggregation Control Protocol (LACP) enables you to bundle several physical ports together to form a single logical channel. LACP enables you to form a single Layer 2 link automatically from two or more Ethernet links. This protocol ensures that both ends of the Ethernet link are functional and are part of the aggregation group.



LACP uses the following parameters to control aggregation:

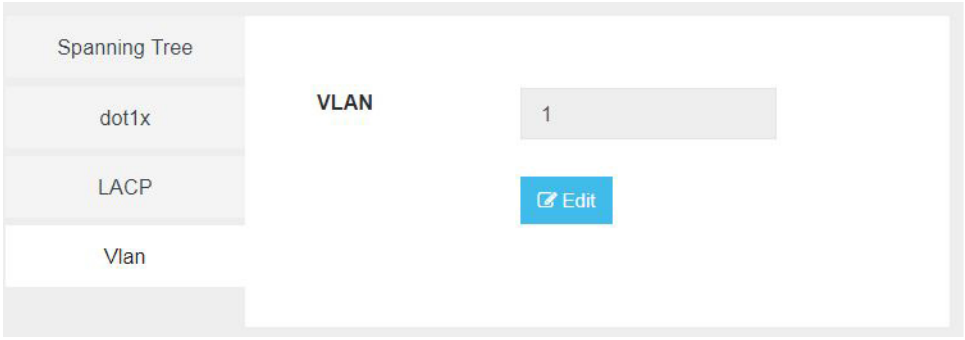
Table 12: LACP Parameters

Parameter	Description	Values
System Priority	Specifies the port priority.	Range: 1 to 65535
Port-channel load balance	Specifies the load balance of the port channel.	Mac Based or IP Based

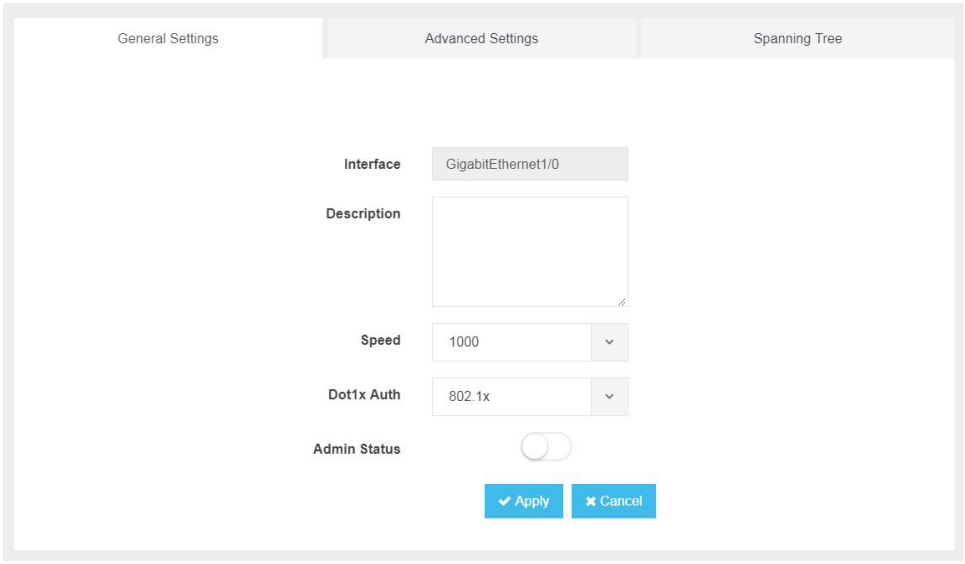
Configuring VLAN

You can use virtual LANs (VLANs) to divide the network into separate logical areas. VLANs can also be considered as broadcast domains. Any switch port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in that VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router.

You can configure VLANs in the range **<1-2349>|<2450-4093>** for a specified switch port.



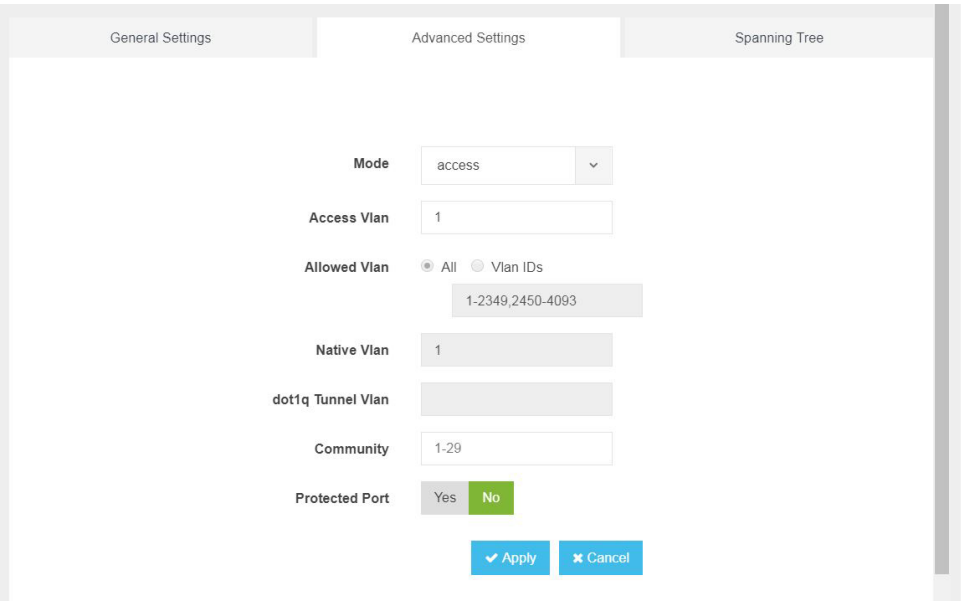
Configuring General Settings



You can configure general settings using the following parameters for each switch interface:

- Interface—Name of the interface
- Description—Set the description per interface
- Speed—10/100/1000 MBPS
- Dot1x Auth—802.1x, mac or both
- PoE Method—auto, never or four-pair
- PoE Limit—0-60000mW
- Admin Status—enable or disable

Configuring Advanced Settings

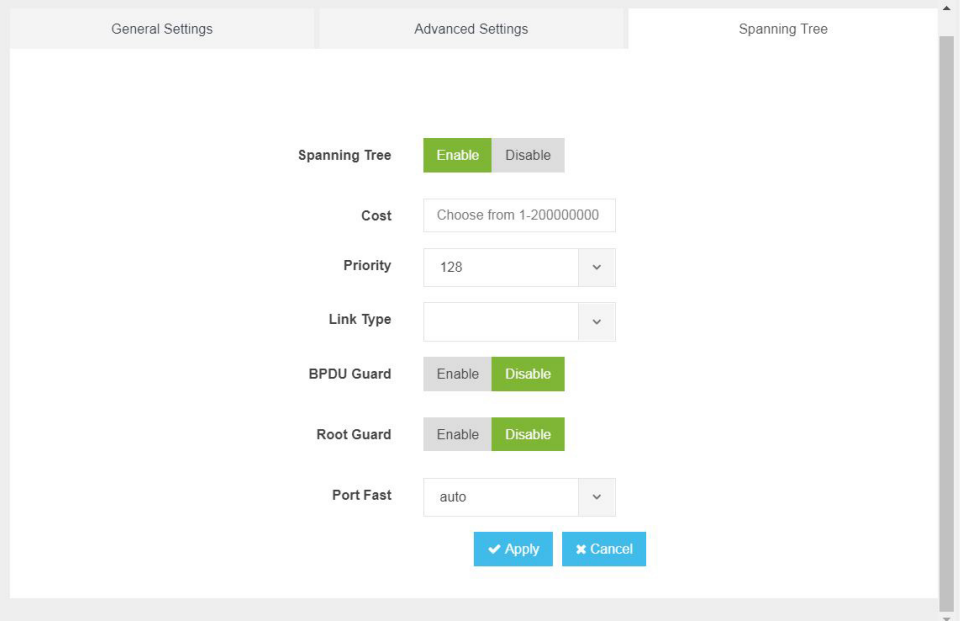


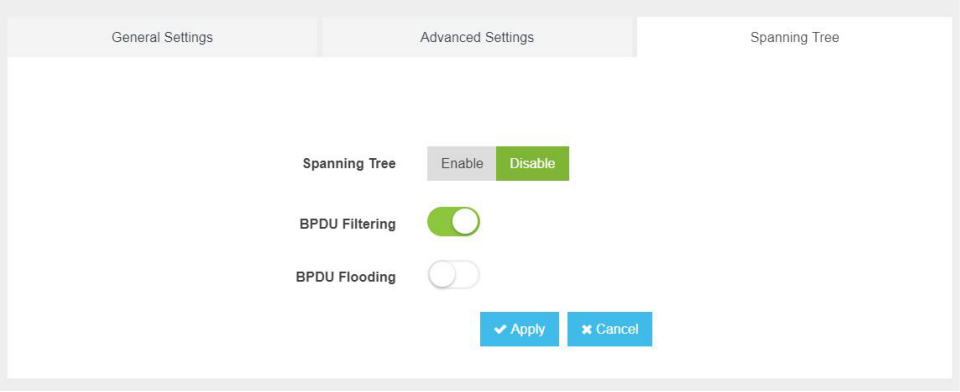
You can make the advanced settings using the following parameters for each switch interface:

- Mode—access, dot1q-tunnel, private-vlan, or trunk
- Access Vlan—Specifies the number of VLANs.
- Allowed Vlan—All or VLAN IDs
- Native Vlan—Specifies the VLAN ID. You can enter a value from one of the following ranges:
  - 1 to 2349
  - 2450 to 4093
- Dot1q Tunnel Vlan—Specifies the Layer 2 tunnel port.
- Community—Specifies the community number. Range: 1 to 29
- Protected Port—Yes or No

Note The VLAN configuration takes effect only if the global VLANs are also configured with the same values in Configuring VLAN, on page 160.

Configuring Spanning Tree per Interface





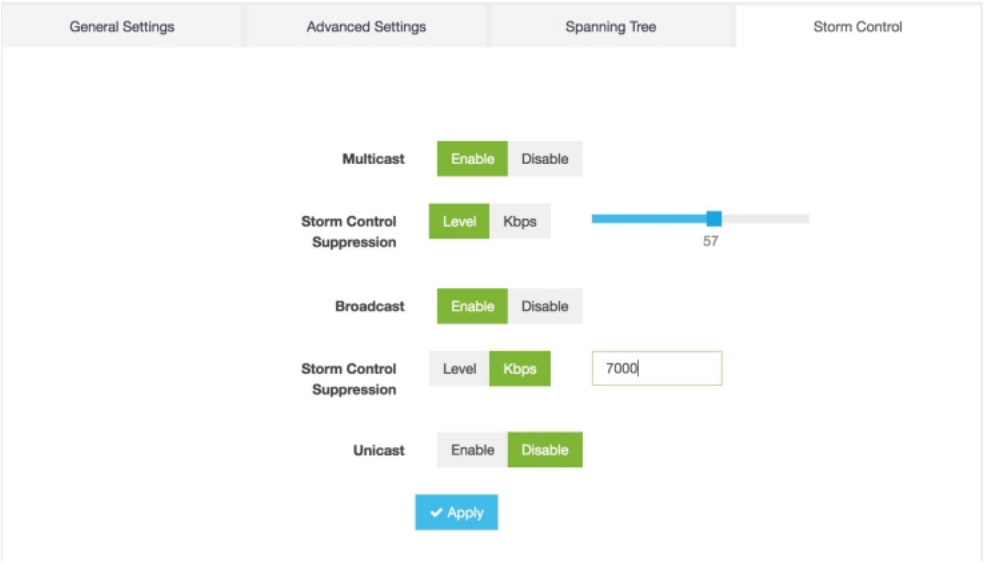
You can configure spanning tree for each switch interface using the following parameters:

- Spanning Tree—Enable or Disable
- Cost—Specifies the cost. Range: 1 to 200000000
- Priority—Specifies the port priority. Range: 0 to 240, default value is 128
- Link Type—point-to-point or shared
- BPDU Guard—Enable or Disable
- Root Guard—Enable or Disable
- Port Fast—auto or enable
- BPDU Filtering—Specifies that BPDU packets are filtered when the spanning tree is disabled
- BPDU Flooding—Specifies that BPDU packets are flooded when the spanning tree is disabled

Configuring Storm Control

Storm control is used to monitor incoming traffic levels and limit excessive flow of packets on any user facing switch port that could cause a traffic storm. Traffic storms can lead to device instability and unintended behavior.

You can configure storm control from NFVIS Portal, from Storm Control tab.



Storm control can be configured for specific type of traffic - unicast or multicast or broadcast. The suppression range can be in terms of a percentage level (1-100) or Kbps value (1-1000000).

Configuring vBranch High Availability

High availability design provides redundancy for WAN, LAN, ENCS device, vRouter, vFirewall VNF level redundancy.

A branch site can have two routers for redundancy. If vEdge-cloud router is chosen, Each of the vedge-cloud router maintains:

- A secure control plane connection, via a DTLS connection, with each vSmart controller in its domain
- A secure data plane connection with the other vEdge routers at the site

Because both vEdge routers receive the same routing information from the vSmart controllers, each one is able to continue to route traffic if one should fail, even if they are connected to different transport providers.

Two firewalls are placed in a group and their configuration is synchronized to prevent a single point of failure on your network. A heartbeat connection between the firewall peers ensures seamless failover in the event that a peer goes down. Setting up two firewalls in an HA pair provides redundancy and allows you to ensure business continuity.

Prerequisites for vBranch HA

The WAN links are active on both Cisco ENCS1 and Cisco ENCS2. Each of the ENCS WAN link is connected to the WAN network (most cases with two SPs), with two ENCSs in an active-active mode.

The LAN facing links of both Cisco ENCS devices are connected to an external switch (as an uplink), and all the devices on the LAN segment are also connected to the external switch. There should be no LAN device connecting directly to the Cisco ENCS internal switch.

Two vRouters and the Two vFirewalls have full mesh L3 connectivity.

VMs and VNFs on both ENCS devices must be configured identical.

SD-Branch HA Design and Topology

In HA design, there are two sets of VLANs. Traffic path is between the VNFs and traffic from or towards LAN.

To protect against cable connection issue and box failure, there is back-to-back cable between ENCS and connection from each ENCS to the external switch.

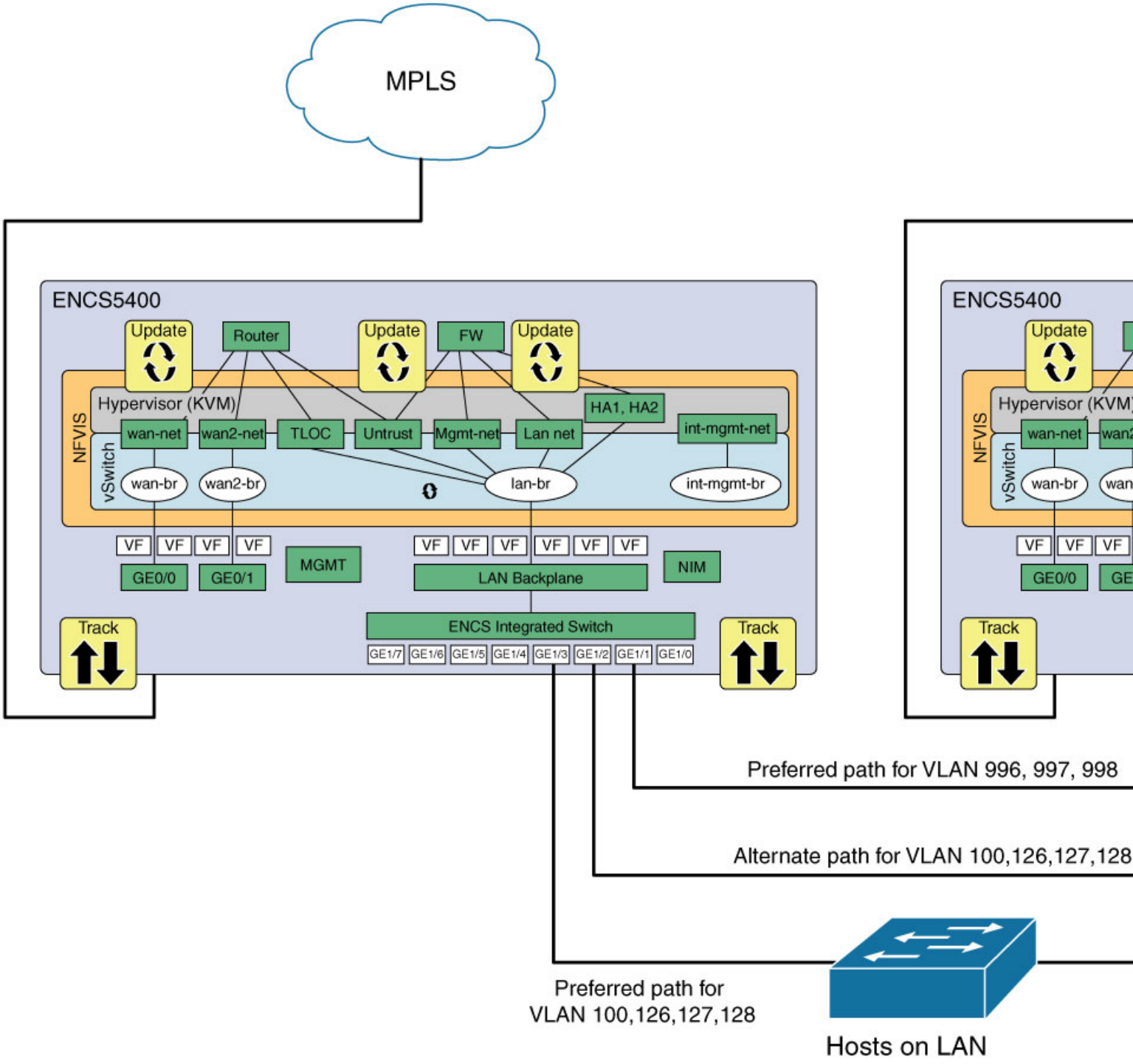
When using Cisco ENCS and Cisco switches, common expectation is to use PVST+, detect loops and switch specific ports to BLOCKING mode. ENCS switch does not support PVST (Per VLAN spanning tree). By Default, RSTP could end up blocking ENCS port back-to-back connection, this will result in blocking traffic path between the VNFs.

The recommended solution is to use MSTP in ENCS and the external switches. The following topology and configuration provides a step-by-step procedure with reasoning for specific configuration use. There are two instances of MSTP created. One for handling traffic path between VNFs and the second for handling traffic from or towards LAN.

Note In cases where external switch cannot be configured for MSTP, RSTP is used and the two links back-2-back between ENCS is not in port-channel.

- One of the links carries traffic between VNFs by configuring disable spanning tree. The second back-to-back link between ENCS processes RSTP and forward or block for the traffic from or towards LAN.
- From each of the ENCS, a third physical link connects to the external switch. This also forwards or blocks the traffic from or towards LAN depending on the RSTP decisions.

Physical Device Connections



VM and Service Chain Network Connection

Figure 6: ENCS-Left

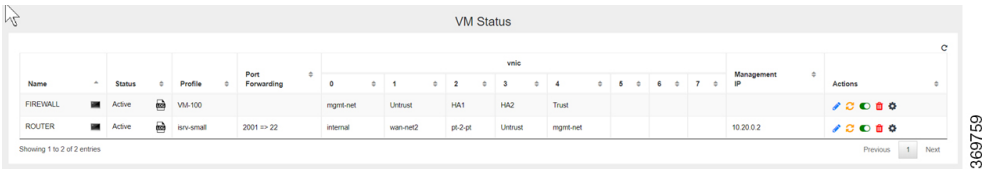
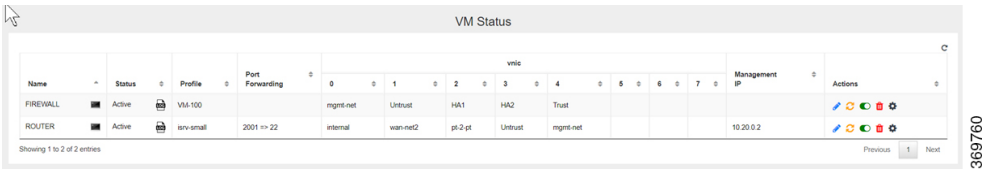


Figure 7: ENCS-Right



Note In the absence of firewall in the design, the router is directly connected to the LAN side. Pt-to-Pt network extends the TLOC connection across the ENCS devices and VRRP is enabled in the router LAN facing connection.

Isolating LAN and Transit Link Traffic for vBranch HA

Traffic from or towards LAN and traffic between the VNFs are isolated by configuring different VLANs for each traffic since both links are connected to the same ENCS internal switch. If you do not isolate the traffic, both LAN traffic and transit link will flow through the same internal switch on the Cisco ENCS.

Enable Port Tracking and Virtual NIC Update

The configured VNICs tracks the state of the ports based on the PNICs notifications. To verify the state of the port, use **show interface** or **ethtool** commands. You can also use commands specific to the VM, that displays the interface link state.

To configure track state on GE0-0 & GE0-1:

```
configure terminal
pnic GE0-0 track-state ROUTER 1
end
```

```
ENCS-Left# support show ifconfig GE0-0

GE0-0: flags=4611<UP,BROADCAST,ALLMULTI,MULTICAST>  mtu 9216
        ether 70:db:98:c3:df:28  txqueuelen 1000  (Ethernet)
```

To configure track state on switch port:

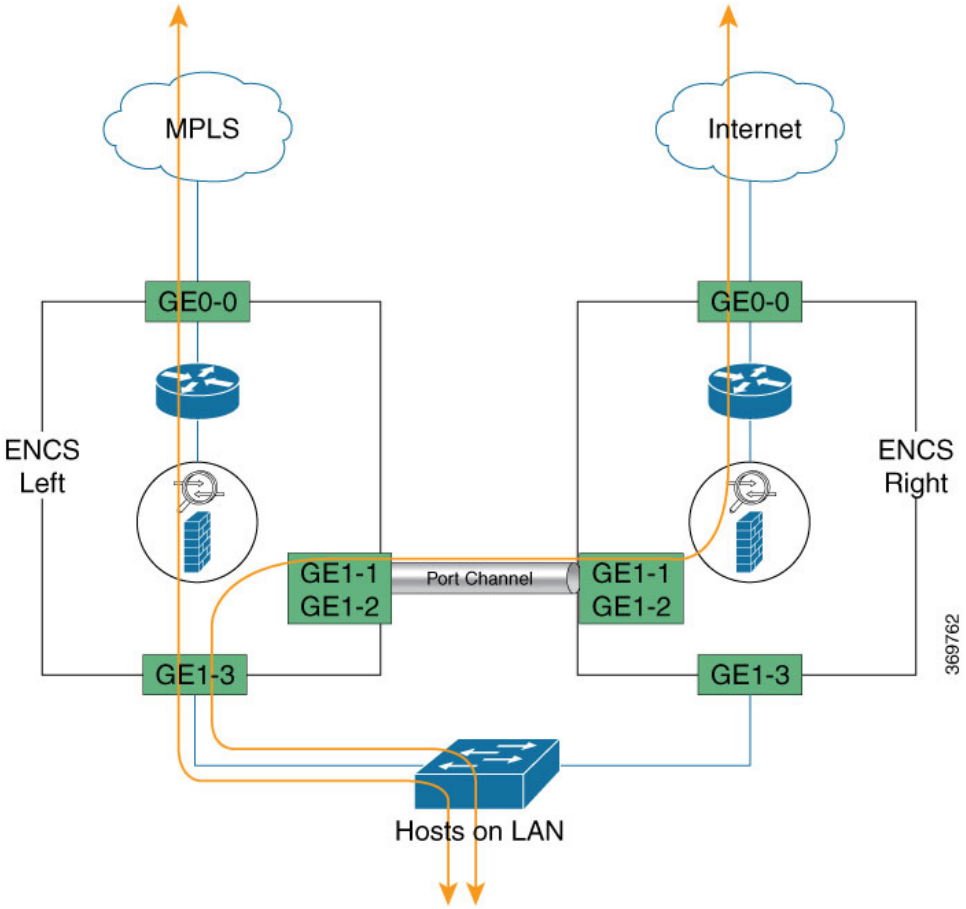
```
configure terminal
switch interface gigabitEthernet 1/3 track-state FIREWALL 4
end
```

```
ENCS-Left# show vm_lifecycle deployments FIREWALL


Name: FIREWALL
Deployment Name : FIREWALL
VM Group Name : FIREWALL
State: ALIVE
Internal State: VM_INERT_STATE
Bootup Time: -1
Image: Palo-Alto-8.1.3.tar.gz
Flavor: VM-100


VCPU#  Memory(MB)  Disk(MB)
----------------------------
2      7168        61440


Low Latency: true
VCPU  CPU  CORE  SOCKET
----------------------
0     3      3     0
1     2      2     0


NICID  VNIC   NETWORK  IP   MAC-ADDRESS        MODEL    PORT-FORWARD
-------------------------------------------------------------------------------
0      vnic6  mgmt-net -    52:54:00:2b:72:d2  virtio
1      vnic7  Untrust  -    52:54:00:eb:a3:e7  virtio
2      vnic8  HA1      -    52:54:00:f4:de:e5  virtio
3      vnic9  HA2      -    52:54:00:12:f8:21  virtio
4      vnic10  Trust    -    52:54:00:7a:6b:e9  virtio



ENCS-Left# support show ifconfig vnic10

vnic10: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9216
        inet6 fe80::fc54:ff:fe7a:6be9  prefixlen 64  scopeid 0x20<link>
        ether fe:54:00:7a:6b:e9  txqueuelen 4000  (Ethernet)
```

Packet Flow for SD-Branch HA

This section explains high-level packet flow in non-failure and failure cases.

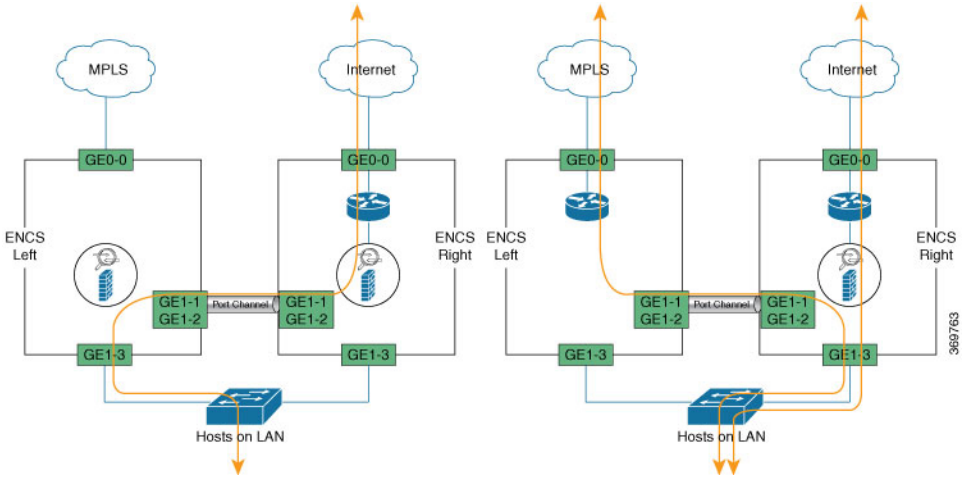
Non-Failure Case



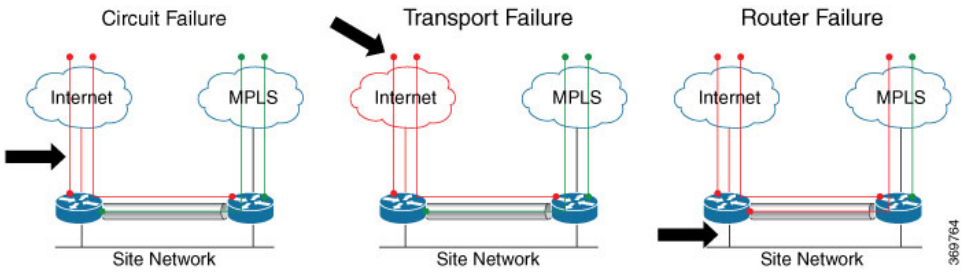
In the non-failure case, both ENCS devices are Active, up and running

- LAN to WAN through the ENCS1 Firewall and ENCS1 Router
- LAN to WAN through the ENCS1 Firewall and ENCS2 Router
- WAN to LAN through ENCS1 Router and ENCS1 Firewall
- WAN to LAN through ENCS2 Router and ENCS1 Firewall

Failure Case



Following are failures that a router must be designed and configured to adapt



The conditions that trigger a firewall failover are:

- One or more of the monitored interfaces fail. (Link Monitoring)
- One or more of the destinations specified on the firewall cannot be reached. (Path Monitoring)
- The firewall does not respond to heartbeat polls. (Heartbeat Polling and Hello messages)

Configuration Examples and Usage Description

ENCS-Left and ENCS-Right with Same Config	Description or Reasons for configuration
networks network wan-net bridge wan-br ! networks network HA1 vlan [126] trunk false bridge lan-br ! networks network HA2 vlan [127] trunk false bridge lan-br ! networks network Trust vlan [128] bridge lan-br ! networks network Untrust vlan [998] bridge lan-br ! networks network mgmt-net vlan [100] trunk false bridge lan-br ! networks network pt-2-pt vlan [996 997] bridge lan-br	In a HA design involving a router or Firewall, there are 3 to 6 paths required. ENCS platform has 2 WAN facing ports and 8 LAN facing ports. • WAN facing ports are reserved for connection to WAN circuits. • LAN facing ports are the only set of available ports for creating the 3 to 6 path required. Between VNFs and LAN, OVS or SR-IOV VFs and physical switch ports are the two Layer2 entities to traverse.
! vlan 1 ! vlan 100 ! vlan 126 ! vlan 127 ! vlan 128 ! vlan 996 ! vlan 997 ! vlan 998 ! spanning-tree enable spanning-tree mode mst spanning-tree mst 2 priority 61440 spanning-tree mst configuration name mst_LAN instance 1 vlan 996-998 instance 2 vlan 100,126-128 !	VLAN must be explicitly created before they are used in the interfaces. Enable MSTP. For MST group 2 carrying “Traffic towards/from LAN”, force the External Switch to become the ROOT using the “mst <group> priority <value>” CLI. The Higher the value, lower the chance of becoming spanning-tree ROOT. “priority” configuration is NOT required for the MST group 1 carrying “Traffic between VNFs”. There is NO loop possibility for MST group 1 VLANs.

ENCS-Left and ENCS-Right with Same Config	Description or Reasons for configuration

nfvis# show running-config switch
switch
interface gigabitEthernet1/1
  no shutdown
  channel-group 1 mode auto
 !
 interface gigabitEthernet1/2
  no shutdown
  channel-group 1 mode auto
!
switch
 interface port-channel1
  negotiation auto
  no shutdown
  spanning-tree mst 1 cost 200000000
  spanning-tree mst 2 cost 200000000
  switchport mode trunk
  switchport trunk native vlan 1
  switchport trunk allowed vlan
100,126-128,996-998
 !
	For the back-to-back ENCS connection, link redundancy is achieved using port-channel configuration. Interfaces that are belong to a port-channel group use configuration from “interface port-channel x”
	
	Goal is to prefer the direct links from ENCS to the External Switch for “Traffic towards/from LAN”. In ENCS back-to-back connection, Spanning tree cost is HIGH for MST group carrying “Traffic towards/from LAN”. This config will block one of the ENCS back-to-back interfaces for breaking the loop for MST group carrying “Traffic towards/from LAN”.

Status of MST instances.

For MST instance 1, “Traffic between the VNFs”, back-to-back portchannel link is root and forwading state.

For MST instance 2, “Traffic from/towards the LAN”, links connected to External Switch are in forwarding state, path via back-to-back portchannel link is “Blocking state”. If one of the Links fail between ENCS and External switch, portchannel path for MST instance 2 will be unblocked.

```
ENCS-Left# show switch vlan detailed

VLAN                TAGGED
ID    VLAN  NAME    PORTS         UNTAGGED PORTS
       CREATED BY
-------------------------------------------------------------------------------
1     1     1     None
gi0,gi4-6,te2,po2-4   DefaultVoiceVLAN
100   100   100   gi3,te2,po1  gi7
         Manual
126   126   126   gi3,te2,po1  None
         Manual
127   127   127   gi3,te2,po1  None
         Manual
128   128   128   gi3,te2,po1  None
         Manual
996   996   996   te2,po1      None
         Manual
997   997   997   te2,po1      None
         Manual
998   998   998   te2,po1      None
         Manual

ENCS-Left# show switch spanning-tree mstp
summary

spanning-tree mstp summary ist-info summary
admin-status enabled
spanning-tree mstp summary ist-info summary
Operation-mode MSTP
spanning-tree mstp summary ist-info summary
Port-Cost-Method long
spanning-tree mstp summary ist-info summary
Loopback-guard disabled
spanning-tree mstp summary ist-info root
Priority 32768
spanning-tree mstp summary ist-info root
Address 70:db:98:c3:df:14
spanning-tree mstp summary ist-info root Cost
 0
spanning-tree mstp summary ist-info root Port
 LAG1
spanning-tree mstp summary ist-info root
Hello-Time 2
spanning-tree mstp summary ist-info root
Max-Age 20
spanning-tree mstp summary ist-info root
Forward-Delay 15
spanning-tree mstp summary ist-info bridge
Priority 32768
spanning-tree mstp summary ist-info bridge
Address 70:db:98:c3:df:a0
spanning-tree mstp summary ist-info bridge
Hello-Time 2
spanning-tree mstp summary ist-info bridge
Max-Age 20
spanning-tree mstp summary ist-info bridge
Forward-Delay 15
spanning-tree mstp summary ist-info
…..
…..

INSTANCE  PRIORITY  DSG ROOT ADDRESS   BRIDGE
```

```
ENCS-Right# show switch vlan detail

VLAN                TAGGED
ID    VLAN  NAME    PORTS         UNTAGGED PORTS
       CREATED BY
-------------------------------------------------------------------------------
1     1     1     None
gi0,gi4-6,te2,po2-4   DefaultVoiceVLAN
100   100   100   gi3,te2,po1  gi7
         Manual
126   126   126   gi3,te2,po1  None
         Manual
127   127   127   gi3,te2,po1  None
         Manual
128   128   128   gi3,te2,po1  None
         Manual
996   996   996   te2,po1      None
         Manual
997   997   997   te2,po1      None
         Manual
998   998   998   te2,po1      None
         Manual

ENCS-Right# show switch spanning-tree mstp
summary

spanning-tree mstp summary ist-info summary
admin-status enabled
spanning-tree mstp summary ist-info summary
Operation-mode MSTP
spanning-tree mstp summary ist-info summary
Port-Cost-Method long
spanning-tree mstp summary ist-info summary
Loopback-guard disabled
spanning-tree mstp summary ist-info root
Priority 32768
spanning-tree mstp summary ist-info root
Address 70:db:98:c3:df:14
spanning-tree mstp summary ist-info root Cost
 0
spanning-tree mstp summary ist-info root Port
 0
spanning-tree mstp summary ist-info root
Hello-Time 2
spanning-tree mstp summary ist-info root
Max-Age 20
spanning-tree mstp summary ist-info root
Forward-Delay 15
spanning-tree mstp summary ist-info bridge
Priority 32768
spanning-tree mstp summary ist-info bridge
Address 70:db:98:c3:df:14
spanning-tree mstp summary ist-info bridge
Hello-Time 2
spanning-tree mstp summary ist-info bridge
Max-Age 20
spanning-tree mstp summary ist-info bridge
Forward-Delay 15
spanning-tree mstp summary ist-info
……
……

INSTANCE  PRIORITY  DSG ROOT ADDRESS   BRIDGE
```

```
 ADDRESS
-------------------------------------------------------------------------------
1          32768     70:db:98:c3:df:14
70:db:98:c3:df:a0
2          61440     f0:b2:e5:56:e4:80
70:db:98:c3:df:a0

INST                   PRIO.
ID    PORT   STATE    NBR    COST     STS
      ROLE
-------------------------------------------------------------------------------
1     gi1/0  enabled  128.1  2000000  disabled
    disabled
1     gi1/3  enabled  128.4  20000
forwarding  designated
1     gi1/4  enabled  128.5  2000000  disabled
    disabled
1     gi1/5  enabled  128.6  2000000  disabled
    disabled
1     gi1/6  enabled  128.7  2000000  disabled
    disabled
1     gi1/7  enabled  128.8  2000000  disabled
    disabled
2     gi1/0  enabled  128.1  2000000  disabled
    disabled
2     gi1/3  enabled  128.4  20000
forwarding  root
2     gi1/4  enabled  128.5  2000000  disabled
    disabled
2     gi1/5  enabled  128.6  2000000  disabled
    disabled
2     gi1/6  enabled  128.7  2000000  disabled
    disabled
2     gi1/7  enabled  128.8  2000000  disabled
    disabled
INST                  PRIO.
ID    PORT  STATE    NBR       COST       STS
         ROLE
-------------------------------------------------------------------------------
1     po1   enabled  128.1000  10000
forwarding  root
1     po2   enabled  128.1001  2000000
disabled    disabled
1     po3   enabled  128.1002  2000000
disabled    disabled
1     po4   enabled  128.1003  2000000
disabled    disabled
2     po1   enabled  128.1000  200000000
blocking    alternate
2     po2   enabled  128.1001  2000000
disabled    disabled
2     po3   enabled  128.1002  2000000
disabled    disabled
2     po4   enabled  128.1003  2000000
disabled    disabled

ENCS-Left#
```

```
 ADDRESS
-------------------------------------------------------------------------------
1          32768     70:db:98:c3:df:14
70:db:98:c3:df:14
2          61440     f0:b2:e5:56:e4:80
70:db:98:c3:df:14

INST                   PRIO.
ID    PORT   STATE    NBR    COST     STS
      ROLE
-------------------------------------------------------------------------------
1     gi1/0  enabled  128.1  2000000  disabled
    disabled
1     gi1/3  enabled  128.4  20000
forwarding  designated
1     gi1/4  enabled  128.5  2000000  disabled
    disabled
1     gi1/5  enabled  128.6  2000000  disabled
    disabled
1     gi1/6  enabled  128.7  2000000  disabled
    disabled
1     gi1/7  enabled  128.8  2000000  disabled
    disabled
2     gi1/0  enabled  128.1  2000000  disabled
    disabled
2     gi1/3  enabled  128.4  20000
forwarding  root
2     gi1/4  enabled  128.5  2000000  disabled
    disabled
2     gi1/5  enabled  128.6  2000000  disabled
    disabled
2     gi1/6  enabled  128.7  2000000  disabled
    disabled
2     gi1/7  enabled  128.8  2000000  disabled
    disabled

INST                  PRIO.
ID    PORT  STATE    NBR       COST       STS
         ROLE
-------------------------------------------------------------------------------
1     po1   enabled  128.1000  10000
forwarding  designated
1     po2   enabled  128.1001  2000000
disabled    disabled
1     po3   enabled  128.1002  2000000
disabled    disabled
1     po4   enabled  128.1003  2000000
disabled    disabled
2     po1   enabled  128.1000  200000000
forwarding  designated
2     po2   enabled  128.1001  2000000
disabled    disabled
2     po3   enabled  128.1002  2000000
disabled    disabled
2     po4   enabled  128.1003  2000000
disabled    disabled
ENCS-Right#
```

From the above summary output, MST instances indicates ID and associated VLAN, and then displays all interfaces as part of VLAN instances. This behaviour differs from the way MST instances are displayed on other Cisco switching platforms.

External Switch MST Configuration

Note It is recommended that VLAN 996-998 is not allowed through the interfaces connecting to ENCS-Left and ENCS-Right. As a result, the external switch MSTP does not participate for VLAN 996-998.

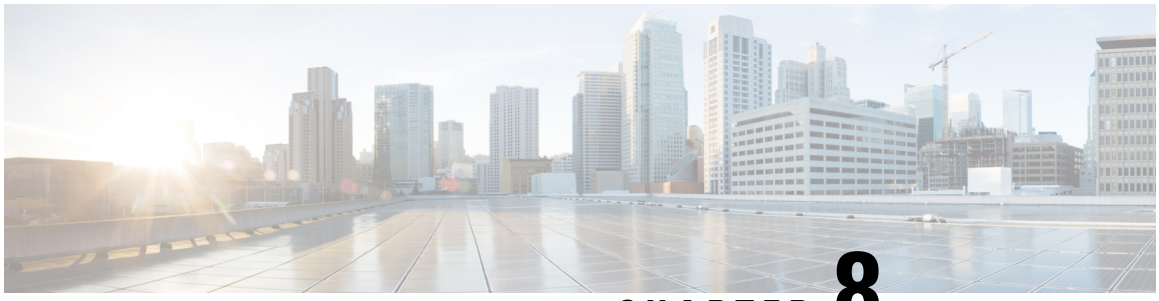
Table 13:

`vlan 100,126-128`
`!`
`spanning-tree mode mst`
`spanning-tree extend system-id`
`spanning-tree uplinkfast`
`!`
`spanning-tree mst configuration`
` name mst_LAN`
` instance 1 vlan 996-998`
` instance 2 vlan 100, 126-128`
`!`
`interface GigabitEthernet1/0/1`
` switchport trunk allowed vlan 100,126-128`
` switchport mode trunk`
`!`
`interface GigabitEthernet1/0/2`
` switchport trunk allowed vlan 100,126-128`
` switchport mode trunk`
	VLANs carrying “Traffic between the VNFs” are NOT sent to the External Switch.
	MST instance priority and MST link COST are kept default in the External Switch.
	MST Priority and COST Configuration in ENCS ensure the External switch is the root and the Interfaces in the External switch connecting to ENCS are in Forwarding state.

Note VLANs carrying traffic between VNFs are not used in external switch and not configured in any interface.

```
Switch#show spanning-tree mst detail

##### MST0    vlans mapped:   1-99,101-125,129-995,999-4094
Bridge        address f0b2.e556.e480  priority      32768 (32768 sysid 0)
Root          address 70db.98c3.df14  priority      32768 (32768 sysid 0)
              port    Gi1/0/2         path cost     0
Regional Root address 70db.98c3.df14  priority      32768 (32768 sysid 0)
                                      internal cost 20000     rem hops 19
Operational   hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured    hello time 2 , forward delay 15, max age 20, max hops    20

GigabitEthernet1/0/1 of MST0 is alternate blocking
Port info             port id          128.1  priority    128  cost       20000
Designated root       address 70db.98c3.df14  priority  32768  cost           0
Design. regional root address 70db.98c3.df14  priority  32768  cost       10000
Designated bridge     address 70db.98c3.dfa0  priority  32768  port id    128.4
Timers: message expires in 5 sec, forward delay 0, forward transitions 0
Bpdus sent 27905, received 31061

GigabitEthernet1/0/2 of MST0 is root forwarding
Port info             port id          128.2  priority    128  cost       20000
Designated root       address 70db.98c3.df14  priority  32768  cost           0
Design. regional root address 70db.98c3.df14  priority  32768  cost           0
Designated bridge     address 70db.98c3.df14  priority  32768  port id    128.4
Timers: message expires in 5 sec, forward delay 0, forward transitions 1
Bpdus sent 27904, received 31070
```

```
##### MST2    vlans mapped:   100,126-128
Bridge        address f0b2.e556.e480  priority      32770 (32768 sysid 2)
Root          this switch for MST2

GigabitEthernet1/0/1 of MST2 is designated forwarding
Port info             port id          128.1  priority    128  cost       20000
Designated root       address f0b2.e556.e480  priority  32770  cost           0
Designated bridge     address f0b2.e556.e480  priority  32770  port id    128.1
Timers: message expires in 0 sec, forward delay 0, forward transitions 1
Bpdus (MRecords) sent 27905, received 31061

GigabitEthernet1/0/2 of MST2 is designated forwarding
Port info             port id          128.2  priority    128  cost       20000
Designated root       address f0b2.e556.e480  priority  32770  cost           0
Designated bridge     address f0b2.e556.e480  priority  32770  port id    128.2
Timers: message expires in 0 sec, forward delay 0, forward transitions 1
Bpdus (MRecords) sent 27904, received 31070

Switch#
```



CHAPTER 8

NFVIS Logging

Configuring System Logs

NFVIS generates log files for troubleshooting issues. The configuration log and the operational log are the two main system log files. The configuration log has information related to configurations and actions performed on the system such as creation of networks. The operational log has information related to system operation such as statistics collection and monitoring.

Log entries can be one of the following types:

Log Level	Purpose
DEBUG	Information, typically of interest only when diagnosing problems.
INFO	Confirmation that things are working as expected.
WARNING	An indication that something unexpected happened, or indicative of some problem in the near future (for example, ‘disk space low’). The software application is still working as expected.
ERROR	Due to a serious problem, the software application is not able to perform some function.
CRITICAL	A serious error, indicating that the program itself may not be able to continue running.

By default, the configuration log has a log-level of INFO. All logs of type INFO, WARNING, ERROR and CRITICAL are logged.

By default, the operational log has a log-level of WARNING. All logs of type WARNING, ERROR and CRITICAL are logged.

The log-level for these log files can be changed using the **system set-log** command:

```
system set-log level error logtype configuration
```

The change to the log level is not persistant across a reboot. After a reboot, the default log levels are used.

The current log files are kept in the */var/log* directory in the system:

- show log - To display the list of available log files
- show log {filename} - To display the contents of a specific log file

Log Rotation

There is a size limit for the log files, under */var/log/* directory. When the log files reach the size limit, the location of logs is rotated to another place. The space limit for the total size of all rotated log files is 2 GB. The older log files are dropped automatically on reaching the space limit. You can also execute a command to trigger the log rotation procedure. The log files are monitored periordically and if a log file gets too big, it is rotated to another place.

There is a size limit for the log files stored in the */var/log* directory. The size of the log files is monitored periodically every fifteen minutes and if a log file gets too big, it is rotated to the */data/intdatastore/logs* directory. The space limit for the total size of all the rotated log files is 2 GB. The older log files are dropped automatically on reaching the space limit. You can also execute the **logrotate** command to trigger the log rotation procedure.

```
nfvis# logrotate
```

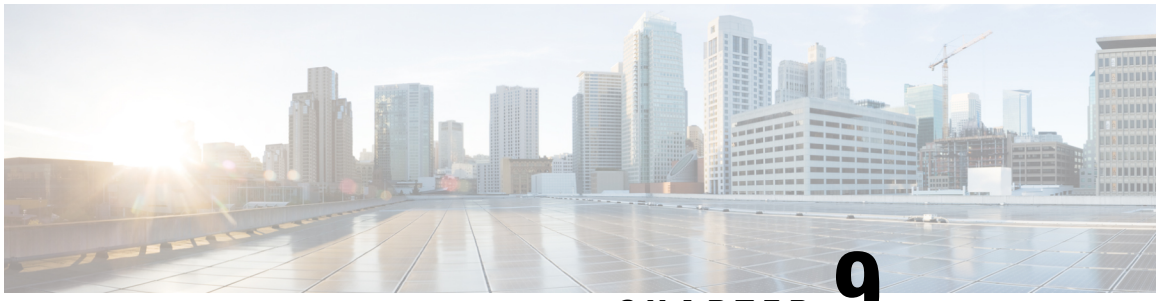
Verifying the System Log Configuration

To verify the system log configuration, use the **show system logging-level** command as shown below:

```
nfvis# show system logging-level
system logging-level configuration error
system logging-level operational warning
```

System Log APIs and Commands

System Log APIs	System Log Commands
• /api/operations/system/set-log • /api/operational/system/logging-level	• system set-log logtype [all/configuration/operational] level [critical/debug/error/info/warning] • show system logging-level



CHAPTER 9

NFVIS Monitoring

Syslog

The Syslog feature allows event notifications from NFVIS to be sent to remote syslog servers for centralized log and event collection.The syslog messages are based on the occurrence of specific events on the device and provide configuration and operational information such as creation of users, changes to the interface status, and failed login attempts. Syslog data is critical to recording day-to-day events as well as notifying operational staff of critical system alerts.

Cisco enterprise NFVIS sends syslog messages to syslog servers configured by the user. Syslogs are sent for Network Configuration Protocol (NETCONF) notifications from NFVIS.

Syslog Message Format

Syslog messages have the following format:

```
<Timestamp> hostname %SYS-<Severity>-<Event>: <Message>
```

Sample Syslog messages:

```
2017 Jun 16 11:20:22 nfvis %SYS-6-AAA_TYPE_CREATE: AAA authentication type tacacs created
successfully AAA authentication set to use tacacs server
2017 Jun 16 11:20:23 nfvis %SYS-6-RBAC_USER_CREATE: Created rbac user successfully: admin
2017 Jun 16 15:36:12 nfvis %SYS-6-CREATE_FLAVOR: Profile created: ISRv-small
2017 Jun 16 15:36:12 nfvis %SYS-6-CREATE_FLAVOR: Profile created: ISRv-medium
2017 Jun 16 15:36:13 nfvis %SYS-6-CREATE_IMAGE: Image created: ISRv_IMAGE_Test
2017 Jun 19 10:57:27 nfvis %SYS-6-NETWORK_CREATE: Network testnet created successfully
2017 Jun 21 13:55:57 nfvis %SYS-6-VM_ALIVE: VM is active: ROUTER
```

Note To refer to the complete list of syslog messages, see Syslog Messages, on page 237

Configure a Remote Syslog Server

To send syslogs to an external server, configure its IP address or DNS name along with the protocol to send syslogs and the port number on the syslog server.

To configure a remote Syslog server:

```
configure terminal
system settings logging host 172.24.22.186
port 3500
transport tcp
commit
```

Note A maximum of 4 remote syslog servers can be configured. The remote syslog server can be specified using its IP address or DNS name. The default protocol for sending syslogs is UDP with a default port of 514. For TCP, the default port is 601.

Configure Syslog Severity

The syslog severity describes the importance of the syslog message.

To configure syslog severity:

```
configure terminal
system settings logging severity <debug | informational | notice | warning| error| critical
 | alert | emergency>
```

Table 14: Syslog Severity Levels

Severity Level	Description	Numeric Encoding for Severity in the Syslog Message Format
debug	Debug-level messages	7
informational	Informational messages	6
notice	Normal but significant condition	5
warning	Warning conditions	4
error	Error conditions	3
critical	Critical conditions	2
alert	Take action immediately	1
emergency	System is unusable	0

Note By default, the logging severity of syslogs is informational which means all syslogs at informational severity and higher will be logged. Configuring a value for severity will result in syslogs at the configured severity and syslogs which are more severe than the configured severity.

Configure Syslog Facility

The syslog facility can be used to logically separate and store syslog messages on the remote syslog server. For example, syslogs from a particular NFVIS can be assigned a facility of local0 and can be stored and processed in a different directory location on the syslog server. This is useful to separate it from syslogs with a facility of local1 from another device.

To configure syslog facility:

```
configure terminal
system settings logging facility local5
```

Note The logging facility can be changed to a facility from local0 to local7

By default, NFVIS sends syslogs with the facility of local7

Syslog Support APIs and Commands

APIs	Commands
• /api/config/system/settings/logging • /api/operational/system/settings/logging	• system settings logging host • system settings logging severity • system settings logging facility

NETCONF Event Notifications

Cisco Enterprise NFVIS generates event notifications for key events. A NETCONF client can subscribe to these notifications for monitoring the progress of configuration activation and the status change of the system and VMs.

There are two types of event notifications: nfvisEvent and vmlcEvent (VM life cycle event)

To receive event notifications automatically, you can run the NETCONF client, and subscribe to these notifications using the following NETCONF operations:

- --create-subscription=nfvisEvent
- --create-subscription=vmlcEvent

You can view NFVIS and VM life cycle event notifications using the **show notification stream nfvisEvent** and **show notification stream vmlcEvent** commands respectively. For more information see, Event Notifications, on page 207.

SNMP Support on NFVIS

Introduction about SNMP

Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

The SNMP framework has three parts:

- SNMP manager - The SNMP manager is used to control and monitor the activities of network hosts using SNMP.
- SNMP agent - The SNMP agent is the software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems.
- MIB - The Management Information Base (MIB) is a virtual information storage area for network management information, which consists of collections of managed objects.

A manager can send the agent requests to get and set MIB values. The agent can respond to these requests. Independent of this interaction, the agent can send unsolicited notifications (traps or informs) to the manager to notify the manager of network conditions.

SNMP Operations

SNMP applications perform the following operations to retrieve data, modify SNMP object variables, and send notifications:

- SNMP Get - The SNMP GET operation is performed by a Network Management Server (NMS) to retrieve SNMP object variables.
- SNMP Set - The SNMP SET operation is performed by a Network Management Server (NMS) to modify the value of an object variable.
- SNMP Notifications - A key feature of SNMP is its capability to generate unsolicited notifications from an SNMP agent.

SNMP Get

The SNMP GET operation is performed by a Network Management Server (NMS) to retrieve SNMP object variables. There are three types of GET operations:

- GET: Retrieves the exact object instance from the SNMP agent.
- GETNEXT: Retrieves the next object variable, which is a lexicographical successor to the specified variable.
- GETBULK: Retrieves a large amount of object variable data, without the need for repeated GETNEXT operations.

The command for SNMP GET is :

snmpget -v2c -c [community-name] [NFVIS-box-ip] [tag-name, example ifSpeed].[index value]

SNMP Walk

SNMP walk is an SNMP application that uses SNMP GETNEXT requests to query a network entity for a tree of information.

An object identifier (OID) may be given on the command line. This OID specifies which portion of the object identifier space will be searched using GETNEXT requests. All variables in the subtree below the given OID are queried and their values presented to the user.

The command for SNMP walk with SNMP v2 is:

snmpwalk -v2c -c [community-name] [nfvis-box-ip]

```
snmpwalk -v2c -c myUser 172.19.147.115 1.3.6.1.2.1.1
SNMPv2-MIB::sysDescr.0 = STRING: Cisco NFVIS
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.9.12.3.1.3.1291
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (43545580) 5 days, 0:57:35.80
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING:
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 70
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
IF-MIB::ifIndex.1 = INTEGER: 1
IF-MIB::ifIndex.2 = INTEGER: 2
IF-MIB::ifIndex.3 = INTEGER: 3
IF-MIB::ifIndex.4 = INTEGER: 4
IF-MIB::ifIndex.5 = INTEGER: 5
IF-MIB::ifIndex.6 = INTEGER: 6
IF-MIB::ifIndex.7 = INTEGER: 7
IF-MIB::ifIndex.8 = INTEGER: 8
IF-MIB::ifIndex.9 = INTEGER: 9
IF-MIB::ifIndex.10 = INTEGER: 10
IF-MIB::ifIndex.11 = INTEGER: 11
IF-MIB::ifDescr.1 = STRING: GE0-0
IF-MIB::ifDescr.2 = STRING: GE0-1
IF-MIB::ifDescr.3 = STRING: MGMT
IF-MIB::ifDescr.4 = STRING: gigabitEthernet1/0
IF-MIB::ifDescr.5 = STRING: gigabitEthernet1/1
IF-MIB::ifDescr.6 = STRING: gigabitEthernet1/2
IF-MIB::ifDescr.7 = STRING: gigabitEthernet1/3
IF-MIB::ifDescr.8 = STRING: gigabitEthernet1/4
IF-MIB::ifDescr.9 = STRING: gigabitEthernet1/5
IF-MIB::ifDescr.10 = STRING: gigabitEthernet1/6
IF-MIB::ifDescr.11 = STRING: gigabitEthernet1/7
...
SNMPv2-SMI::mib-2.47.1.1.1.1.2.0 = STRING: "Cisco NFVIS"
SNMPv2-SMI::mib-2.47.1.1.1.1.3.0 = OID: SNMPv2-SMI::enterprises.9.1.1836
SNMPv2-SMI::mib-2.47.1.1.1.1.4.0 = INTEGER: 0
SNMPv2-SMI::mib-2.47.1.1.1.1.5.0 = INTEGER: 3
SNMPv2-SMI::mib-2.47.1.1.1.1.6.0 = INTEGER: -1
SNMPv2-SMI::mib-2.47.1.1.1.1.7.0 = STRING: "ENCS5412/K9"
SNMPv2-SMI::mib-2.47.1.1.1.1.8.0 = STRING: "M3"
SNMPv2-SMI::mib-2.47.1.1.1.1.9.0 = ""
SNMPv2-SMI::mib-2.47.1.1.1.1.10.0 = STRING: "3.7.0-817"
SNMPv2-SMI::mib-2.47.1.1.1.1.11.0 = STRING: "FGL203012P2"
SNMPv2-SMI::mib-2.47.1.1.1.1.12.0 = STRING: "Cisco Systems, Inc."
SNMPv2-SMI::mib-2.47.1.1.1.1.13.0 = ""
...
```

The following is a sample configuration of SNMP walk with SNMP v3:

```
snmpwalk -v 3 -u user3 -a sha -A changePassphrase -x aes -X changePassphrase -l authPriv
-n snmp 172.16.1.101 system
SNMPv2-MIB::sysDescr.0 = STRING: Cisco ENCS 5412, 12-core Intel, 8 GB, 8-port PoE LAN, 2
HDD, Network Compute System
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.9.1.2377
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (16944068) 1 day, 23:04:00.68
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING:
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 70
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
```

SNMP Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. Unsolicited (asynchronous) notifications can be generated as traps or inform requests. Traps are messages alerting the SNMP manager to a condition on the network. Inform requests (informs) are traps that include a request for confirmation of receipt from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.

Note Starting from Release 3.8.1 NFVIS has SNMP Trap support for switch interfaces. If a trap server is setup in the NFVIS snmp configuration, it will send trap messages for both NFVIS and switch interfaces. Both the interfaces are triggered by the link state up or down by unplugging a cable or setting admin_state up or down when a cable is connected.

SNMP Versions

Cisco enterprise NFVIS supports the following versions of SNMP:

- SNMP v1—The Simple Network Management Protocol: A Full Internet Standard, defined in RFC 1157. (RFC 1157 replaces the earlier versions that were published as RFC 1067 and RFC 1098.) Security is based on community strings.
- SNMP v2c—The community-string based Administrative Framework for SNMPv2. SNMPv2c (the "c" stands for "community") is an Experimental Internet Protocol defined in RFC 1901, RFC 1905, and RFC 1906. SNMPv2c is an update of the protocol operations and data types of SNMPv2p (SNMPv2 Classic), and uses the community-based security model of SNMPv1.
- SNMPv3—Version 3 of SNMP. SNMPv3 is an interoperable standards-based protocol defined in RFCs 3413 to 3415. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network.

The security features provided in SNMPv3 are as follows:

- Message integrity—Ensuring that a packet has not been tampered with in transit.
- Authentication—Determining that the message is from a valid source.
- Encryption—Scrambling the contents of a packet to prevent it from being learned by an unauthorized source.

Both SNMP v1 and SNMP v2c use a community-based form of security. The community of managers able to access the agent MIB is defined by an IP address Access Control List and password.

SNMPv3 is a security model in which an authentication strategy is set up for a user and the group in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

Authentication of the community with the user configuration is implemented even though SNMP v1 and v2 traditionally do not require a user configuration to be set. For both SNMP v1 and v2 on NFVIS, the user must be set with the same name and version as the corresponding community name. The user group must also match an existing group with the same SNMP version for snmpwalk commands to work.

SNMP MIB Support

The following MIB's are supported for SNMP on NFVIS:

IF-MIB:

- ifDescr
- ifType
- ifPhysAddress
- ifSpeed
- ifOperStatus
- ifAdminStatus
- ifMtu
- ifName
- ifHighSpeed
- ifPromiscuousMode
- ifConnectorPresent
- ifInErrors
- ifInDiscards
- ifInOctets
- ifOutErrors
- ifOutDiscards
- ifOutOctets
- ifOutUcastPkts
- ifHCInOctets
- ifHCInUcastPkts
- ifHCOutOctets

- ifHCOutUcastPkts
- ifInBroadcastPkts
- ifOutBroadcastPkts
- ifInMulticastPkts
- ifOutMulticastPkts
- ifHCInBroadcastPkts
- ifHCOutBroadcastPkts
- ifHCInMulticastPkts
- ifHCOutMulticastPkts

Entity MIB:

- entPhysicalIndex
- entPhysicalDescr
- entPhysicalVendorType
- entPhysicalContainedIn
- entPhysicalClass
- entPhysicalParentRelPos
- entPhysicalName
- entPhysicalHardwareRev
- entPhysicalFirmwareRev
- entPhysicalSoftwareRev
- entPhysicalSerialNum
- entPhysicalMfgName
- entPhysicalModelName
- entPhysicalAlias
- entPhysicalAssetID
- entPhysicalIsFRU

Cisco Process MIB:

- cpmCPUTotalPhysicalIndex
- cpmCPUMonInterval
- cpmCPUMemoryKernelReserved
- cpmCPUMemoryHCKernelReserved

- cpmCPUMemoryUsed
- cpmCPUMemoryFree
- cpmCPUMemoryHCUsed
- cpmCPUMemoryHCFree
- CISCO_ENVMON_MIB
- cpmProcessDynamicMemorySizeOvrflw
- cpmProcessType
- cpmCPULoadAvg1min
- cpmCPULoadAvg5min
- cpmCPULoadAvg15min

Configuring SNMP Support

Though SNMP v1 and v2c uses community-based string, the following is still required:

- Same community and user name.
- Same SNMP version for user and group.

To create SNMP community:

```
configure terminal
snmp community <community_name> community-access <access>
```

SNMP community name string supports [A-Za-z0-9_-] and maximum length of 32. NFVIS supports only **readOnly** access.

To create SNMP Group:

```
configure terminal
snmp group <group_name> <context> <version> <security_level> notify <notify_list> read
<read_list> write <write_list>
```

Variables	Description
group_name	Group name string. Supporting string is [A-Za-z0-9_-] and maximum length is 32.
context	Context string, default is snmp. Maximum length is 32. Minimum length is 0 (empty context).
version	1, 2 or 3 for SNMP v1, v2c and v3.
security_level	authPriv, authNoPriv, noAuthNoPriv **Note** SNMP v1 and v2c uses noAuthNoPriv only.

Variables	Description
notify_list/read_list/write_list	It can be any string. read_list and notify_list is required to support data retrieval by SNMP tools. write_list can be skipped because NFVIS SNMP does not support SNMP write access.

To create SNMP user:

When security level is authPriv

```
configure terminal
snmp user <user_name> user-version <version> user-group <group_name> auth-protocol <auth>
priv-protocol <priv> passphrase <passphrase_string>
```

When security level is authNoPriv:

```
configure terminal
snmp user <user_name> user-version <version> user-group <group_name> auth-protocol <auth>
passphrase <passphrase_string>
```

When security level is noAuthNopriv

```
configure terminal
snmp user <user_name> user-version <version> user-group <group_name>
```

Variables	Description
user_name	User name string. Supporting string is [A-Za-z0-9_-] and maximum length is 32. This name has to be the same as community_name.
version	1 and 2 for SNMP v1 and v2c.
group_name	Group name string. This name has to be same as the group name configured in the NFVIS.
auth	md5 or sha
priv	aes or des
passphrase_string	Passphrase string. Supporting string is [A-Za-z0-9\-_#@%$*&!]. NFVIS does not support different passphrases for auth and priv.

Note Do not use auth-key and priv-key. The auth and priv passphrases are encrypted after configuration and saved in NFVIS.

To enable SNMP traps:

```
configure terminal
snmp enable traps <trap_event>
```

trap_event can be **linkup** or **linkdown**

To create SNMP traps:

```
configure terminal
snmp host <host_name> host-ip-address <ip_address> host-port <port> host-user-name <user_name>
 host-version <version> host-security-level noAuthNoPriv
```

Variables	Description
host_name	User name string. Supporting string is [A-Za-z0-9_-] and maximum length is 32. This is not FQDN host name, but an alias to IP address of traps.
ip_address	IP address of traps server.
port	Default is 162. Change to other port number based on your own setup.
user_name	User name string. Must be the same as user_name configured in NFVIS.
version	1, 2 or 3 for SNMP v1, v2c or v3.
security_level	authPriv, authNoPriv, noAuthNoPriv **Note** SNMP v1 and v2c uses noAuthNoPriv only.

The following example shows SNMP v1 and v2 configuration:

```
configure terminal
snmp community public community-access readOnly
!
snmp group testgroup snmp 2 noAuthNoPriv read read-access write write-access notify
notify-access
!
snmp user public user-group testgroup user-version 2
!
snmp host host2 host-ip-address 2.2.2.2 host-port 162 host-user-name public host-version 2
 host-security-level noAuthNoPriv
!
snmp enable traps linkup
snmp enable traps linkDown
```

The following example shows SNMP v3 configuration:

```
configure terminal
snmp group testgroup3 snmp 3 authPriv notify test write test read test
!
snmp user user3 user-version 3 user-group testgroup3 auth-protocol sha priv-protocol aes
passphrase changePassphrase
! configure snmp host to enable snmp v3 trap
snmp host host3 host-ip-address 3.3.3.3 host-version 3 host-user-name user3
host-security-level authPriv host-port 162
!!
```

To change the security level:

```
configure terminal
!
snmp group testgroup4 snmp 3 authNoPriv notify test write test read test
!
snmp user user4 user-version 3 user-group testgroup4 auth-protocol md5 passphrase
changePassphrase
! configure snmp host to enable snmp v3 trap
snmp host host4 host-ip-address 4.4.4.4 host-version 3 host-user-name user4
host-security-level authNoPriv host-port 162
!!
snmp enable traps linkUp
snmp enable traps linkDown
```

To change default context SNMP:

```
configure terminal
!
snmp group testgroup5 devop 3 authPriv notify test write test read test
!
snmp user user5 user-version 3 user-group testgroup5 auth-protocol md5 priv-protocol des
passphrase changePassphrase
!
```

To use empty context and noAuthNoPriv

```
configure terminal
!
snmp group testgroup6 "" 3 noAuthNoPriv read test write test notify test
!
snmp user user6 user-version 3 user-group testgroup6
!
```

Note SNMP host configuration is supported for NFVIS 3.6.1 release. Host trap server configuration will be officially supported for NFVIS 3.7.1 release.

Note SNMP v3 context **snmp** is added automatically when configured from the web portal. To use a different context value or empty context string, use NFVIS CLI or API for configuration.

NFVIS SNMP v3 only supports single passphrase for both auth-protocol and priv-protocol.

Do not use auth-key and priv-key to configure SNMP v3 passphrase. These keys are generated differently between different NFVIS systems for the same passphrase.

Note NFVIS 3.11.1 release enhances the special character support for passphrase. Now the following characters are supported: @#$-!&*

Note NFVIS 3.12.1 release supports the following special characters: -_#@%$*&! and whitespace. Backslash (\) is not supported.

Verify the configuration for SNMP support

Use the **show snmp agent** command to verify the snmp agent description and ID.

```
nfvis# show snmp agent

snmp agent sysDescr "Cisco NFVIS "
snmp agent sysOID 1.3.6.1.4.1.9.12.3.1.3.1291
```

Use the **show snmp traps** command to verify the state of snmp traps.

```
nfvis# show snmp traps

TRAP      TRAP
NAME      STATE
--------------------
linkDown  disabled
linkUp    enabled
```

Use the **show snmp stats** command to verify the snmp stats.

```
nfvis# show snmp stats

snmp stats sysUpTime    57351917
snmp stats sysServices  70
snmp stats sysORLastChange 0
snmp stats snmpInPkts   104
snmp stats snmpInBadVersions 0
snmp stats snmpInBadCommunityNames 0
snmp stats snmpInBadCommunityUses 0
snmp stats snmpInASNParseErrs 0
snmp stats snmpSilentDrops 0
snmp stats snmpProxyDrops 0
```

Use the **show running-config snmp** command to verify the interface configuration for snmp.

```
nfvis# show running-config snmp

snmp agent enabled true
snmp agent engineID 00:00:00:09:11:22:33:44:55:66:77:88
snmp enable traps linkUp
snmp community pub_comm
community-access readOnly
!
snmp community tachen
community-access readOnly
!
snmp group tachen snmp 2 noAuthNoPriv
read   test
write  test
notify test
```

```
!
snmp group testgroup snmp 2 noAuthNoPriv
read   read-access
write  write-access
notify notify-access
!
snmp user public
user-version  2
user-group    2
auth-protocol md5
priv-protocol des
!
snmp user tachen
user-version 2
user-group   tachen
!
snmp host host2
host-port           162
host-ip-address     2.2.2.2
host-version        2
host-security-level noAuthNoPriv
host-user-name      public
!
```

Upper limit for SNMP configurations:

- Communities: 10
- Groups: 10
- Users: 10
- Hosts: 4

SNMP Support APIs and Commands

APIs	Commands
• /api/config/snmp/agent • /api/config/snmp/communities • /api/config/snmp/enable/traps • /api/config/snmp/hosts • /api/config/snmp/user • /api/config/snmp/groups	• agent • community • trap-type • host • user • group

System Monitoring

NFVIS provides system monitoring commands and APIs to monitor the host and the VMs deployed on NFVIS. These commands are useful to collect statistics on CPU utilization, memory, disk and ports. The metrics related to these resources are collected periodically and displayed for a specified duration. For larger durations average values are displayed.

System monitoring enables the user to view historical data on the system's operation.These metrics are also shown as graphs on the portal.

Collection of System Monitoring Statistics

System monitoring statistics are displayed for the requested duration. The default duration is five minutes.

The supported duration values are 1min, 5min, 15min, 30min, 1h, 1H, 6h, 6H, 1d, 1D, 5d, 5D, 30d, 30D with min as minutes, h and H as hours, d and D as days.

Example

The following is a sample output of system monitoring statistics:

```
nfvis# show system-monitoring host cpu stats cpu-usage 1h state non-idle
system-monitoring host cpu stats cpu-usage 1h state non-idle
  collect-start-date-time  2019-12-20T11:27:20-00:00
  collect-interval-seconds 10
  cpu
     id               0
     usage-percentage "[7.67, 5.52, 4.89, 5.77, 5.03, 5.93, 10.07, 5.49,
     ...
```

The time at which the data collection started is displayed as **collect-start-date-time**.

The sampling interval at which data is collected is shown as **collect-interval-seconds**.

The data for the requested metric like host CPU statistics is displayed as an array. The first data point in the array was collected at the specified **collect-start-date-time** and each subsequent value at an interval specified by **collect-interval-seconds**.

In the sample output, CPU id 0 has a utilization of 7.67% on 2019-12-20 at 11:27:20 as specified by **collect-start-date-time**. 10 seconds later, it had a utilization of 5.52% since the **collect-interval-seconds** is 10. The third value of cpu-utilization is 4.89% at 10 seconds after the second value of 5.52% and so on.

The sampling interval shown as **collect-interval-seconds** changes based on the specified duration. For higher durations, the collected statistics are averaged at a higher interval to keep the number of results reasonable.

Host System Monitoring

NFVIS provides system monitoring commands and APIs to monitor the host's CPU utilization, memory, disk and ports.

Monitoring the Host CPU Usage

The percentage of time spent by the CPU in various states, such as executing user code, executing system code, waiting for IO operations, etc. is displayed for the specified duration.

cpu-state	Description
non-idle	100 – idle-cpu-percentage
interrupt	Indicates the percentage of the processor time spent in servicing interrupts

cpu-state	Description
nice	The nice CPU state is a subset of the user state and shows the CPU time used by processes that have a lower priority than other tasks.
system	The system CPU state shows the amount of CPU time used by the kernel.
user	The user CPU state shows CPU time used by user space processes
wait	Idle time while waiting for an I/O operation to complete

The non-idle state is what the user usually needs to monitor. Use the following CLI or API for monitoring CPU usage:

```
nfvis# show system-monitoring host cpu stats cpu-usage <duration> state <cpu-state>
```

```
/api/operational/system-monitoring/host/cpu/stats/cpu-usage/<duration>,<cpu-state>?deep
```

The data is also available in an aggregate form for the minimum, maximum, and average CPU utilization using the following CLI and API:

```
nfvis# show system-monitoring host cpu table cpu-usage <duration>
```

```
/api/operational/system-monitoring/host/cpu/table/cpu-usage/<duration>?deep
```

Monitoring Host Memory

Statistics for the physical memory utilization are displayed for the following categories:

Field	Description
buffered-MB	Memory used for buffering I/O
cached-MB	Memory used for caching file system access
free-MB	Memory available for use
used-MB	Memory in use by the system
slab-recl-MB	Memory used for SLAB-allocation of kernel objects, that can be reclaimed
slab-unrecl-MB	Memory used for SLAB-allocation of kernel objects, that can’t be reclaimed

Use the following CLI or API for monitoring host memory:

```
nfvis# show system-monitoring host memory stats mem-usage <duration>
```

```
/api/operational/system-monitoring/host/memory/stats/mem-usage/<duration>?deep
```

The data is also available in an aggregate form for the minimum, maximum, and average memory utilization using the following CLI and API:

```
nfvis# show system-monitoring host memory table mem-usage <duration>
```

```
/api/operational/system-monitoring/host/memory/table/mem-usage/<duration>?deep
```

Monitoring Host Disks

Statistics for disk operations and disk space can be obtained for the list of disks and disk partitions on the NFVIS host.

Monitoring Host Disks Operations

The following disk performance statistics are displayed for each disk and disk partition:

Field	Description
io-time-ms	Average time spent doing I/O operations in milliseconds
io-time-weighted-ms	Measure of both I/O completion time and the backlog that may be accumulating
merged-reads-per-sec	The number of read operations that could be merged into already queued operations, that is one physical disk access served two or more logical operations. The higher the merged reads, the better the performance.
merged-writes-per-sec	The number of write operations that could be merged into other already queued operations, that is one physical disk access served two or more logical operations. The higher the merged reads, the better the performance.
bytes-read-per-sec	Bytes read per second
bytes-written-per-sec	Bytes written per second
reads-per-sec	Number of read operations per second
writes-per-sec	Number of write operations per second
time-per-read-ms	The average time a read operation takes to complete
time-per-write-ms	The average time a write operation takes to complete
pending-ops	The queue size of pending I/O operations

Use the following CLI or API for monitoring host disks:

```
nfvis# show system-monitoring host disk stats disk-operations <duration>
```

```
/api/operational/system-monitoring/host/disk/stats/disk-operations/<duration>?deep
```

Monitoring Host Disk Space

The following data related to file system usage, that is how much space on a mounted partition is used and how much is available is collected:

Field	Description
free-GB	Gigabytes available
used-GB	Gigabytes in use
reserved-GB	Gigabytes reserved for the root user

Use the following CLI or API for monitoring host disk space:

```
nfvis# show system-monitoring host disk stats disk-space <duration>
```

```
/api/operational/system-monitoring/host/disk/stats/disk-space/<duration>?deep
```

Monitoring Host Ports

The following statistics for network traffic and errors on interfaces are displayed:

Field	Description
name	Interface name
total-packets-per-sec	Total (received and transmitted) packet rate
rx-packets-per-sec	Packets received per second
tx-packets-per-sec	Packets transmitted per second
total-errors-per-sec	Total (received and transmitted) error rate
rx-errors-per-sec	Error rate for received packets
tx-errors-per-sec	Error rate for transmitted packets

Use the following CLI or API for monitoring host ports:

```
nfvis# show system-monitoring host port stats port-usage <duration>
```

```
/api/operational/system-monitoring/host/port/stats/port-usage/<duration>?deep
```

The data is also available in an aggregate form for the minimum, maximum, and average port utilization using the following CLI and API:

```
nfvis# show system-monitoring host port table
```

```
/api/operational/system-monitoring/host/port/table/port-usage/<duration>,<name>?deep
```

VNF System monitoring

NFVIS provides system monitoring commands and APIs to get statistics on the virtualized guests deployed on NFVIS. These statistics provide data on the VM's CPU utilization, memory, disk and network interfaces.

Monitoring the VNF CPU Usage

The CPU utilization of a VM is displayed for the specified duration using the following fields:

Field	Description
total-percentage	Average CPU utilization across all the logical CPUs used by the VM
id	Logical CPU ID
vcpu-percentage	CPU utilization percentage for the specified logical CPU id

Use the following CLI or API to monitor the CPU usage of the VNF:

```
nfvis# show system-monitoring vnf vcpu stats vcpu-usage <duration>
```

```
/api/operational/system-monitoring/vnf/vcpu/stats/vcpu-usage/<duration>?deep
```

```
/api/operational/system-monitoring/vnf/vcpu/stats/vcpu-usage/<duration>/vnf/<vnf-name>?deep
```

Monitoring VNF memory

The following statistics are collected for VNF memory utilization:

Field	Description
total-MB	Total memory of the VNF in MB
rss-MB	Resident Set Size (RSS) of the VNF in MB The Resident Set Size (RSS) is the portion of memory occupied by a process, that is held in the RAM. The rest of the occupied memory exists in the swap space or file system, because some parts of the occupied memory are paged out, or some parts of the executable are not loaded.

Use the following CLI or API to monitor VNF memory:

```
nfvis# show system-monitoring vnf memory stats mem-usage <duration>
```

```
/api/operational/system-monitoring/vnf/memory/stats/mem-usage/<duration>?deep
```

```
/api/operational/system-monitoring/vnf/memory/stats/mem-usage/<duration>/vnf/<vnf-name>?deep
```

Monitoring VNF Disks

The following disk performance statistics are collected for each disk used by the VM:

Field	Description
bytes-read-per-sec	Bytes read from the disk per second
bytes-written-per-sec	Bytes written to the disk per second
reads-per-sec	Number of read operations per second
writes-per-sec	Number of write operations per second

Use the following CLI or API to monitor VNF disks:

```
nfvis# show system-monitoring vnf disk stats <duration>
```

```
/api/operational/system-monitoring/vnf/disk/stats/disk-operations/<duration>?deep
```

```
/api/operational/system-monitoring/vnf/disk/stats/disk-operations/<duration>/vnf/<vnf-name>?deep
```

Monitoring VNF Ports

The following network interface statistics are collected for VMs deployed on NFVIS:

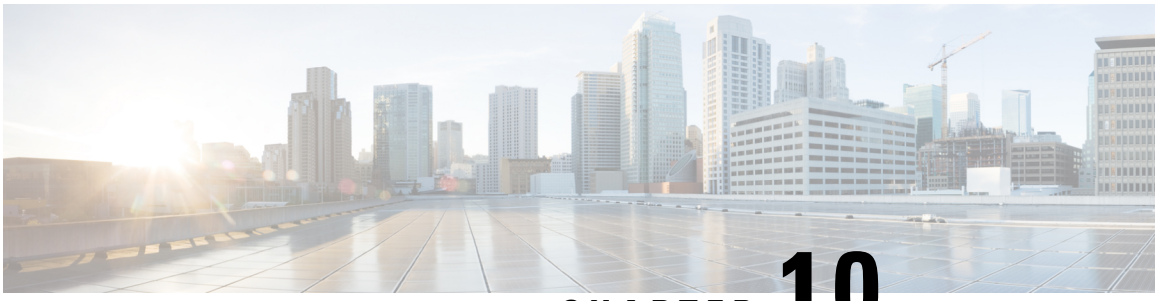
Field	Description
total-packets-per-sec	Total packets received and transmitted per second
rx-packets-per-sec	Packets received per second
tx-packets-per-sec	Packets transmitted per second
total-errors-per-sec	Total error rate for packet reception and transmission
rx-errors-per-sec	Error rate for receiving packets
tx-errors-per-sec	Error rate for transmitting packets

Use the following CLI or API to monitor VNF ports:

```
nfvis# show system-monitoring vnf port stats port-usage <duration>
```

```
/api/operational/system-monitoring/vnf/port/stats/port-usage/<duration>?deep
```

```
/api/operational/system-monitoring/vnf/port/stats/port-usage/<duration>/vnf/<vnf-name>?deep
```



CHAPTER 10

Troubleshoot and Debug Cisco NFVIS

Log and Show Commands

Support Commands and Show Commands

The following commands translate to corresponding linux commands like virsh, ovs and ip:

Command	Description
`show system status`	To display system defaults and services status.
`show system disk-space`	To display information about the system disk space.
`show system memory`	To display information about the system memory. If DPDK is enabled, check if HugePage is available to use.
`show resources cpu-info`	To get information on the resource assignment.
VM	
`support virsh all-info`	To display the output of all supported VM and index by number.
`support virsh dumpxml <num>`	To display all information about one VM index by <num>
`support virsh domiflist <num>`	To display the list of interfaces on VM index by <num> and MAC address of the VNICs.
Network	
`support show ifconfig`	To display the configuration details of all network interfaces or a specific interface.

Command	Description
support virsh net-list	To display all the networks in the host
support virsh net-dumpxml <network name>	To display the network information about one network and bridge attachment.
support virsh iface-list	To display a list of interfaces on the host.
Bridge	
support ovs vsctl show: : :	To display an overview of the bridge, port and vlan tag.
support ovs appctl fdb-show <bridge-name>	To display information about the ports of a bridge.
support ovs all-info	To display the output of all supported ovs commands
Firewall	
support show firewall get-all-rule	

Log Files

The tech-support includes all the logs. Download tech-support and record the time of the occurance of error.

Command	Description
show log	To display a list of available log files or content of a specific log file.
show log nfvis_syslog.log	To display syslogs.
show log nfvis_config.log	To display system configuration related logs.
show log esc/escmanager.log	To display VM deployment related logs.
show log switch_confiog.log	To display the built-in switch configuration logs.

SPAN Session or Port Mirroring

About SPAN Sessions

The Switched Port Analyzer (SPAN) or Port Mirroring feature helps you analyze network traffic passing through interfaces or VLANs by using SPAN sessions. The SPAN sessions send a copy (mirror) of the traffic to another interface or VLAN on the switch that has been connected to a network analyzer or monitoring device. SPAN does not affect the switching of network traffic on the source interfaces.

Note You must dedicate a destination port for SPAN use. Except for traffic that is required for the SPAN session, destination ports do not receive or forward traffic. When the SPAN is configured on the system, there might be some performance hit.

SPAN Session Interfaces

The interface can be:

- Physical interface
- LAN SRIOV
- VM's vNIC (virtio net)

In the case of virtio net or SRIOV VF, you have to specify the VM group name and NIC ID of the VM interface. If the VM vNIC is virtio net type, then the SPAN session is applied on the OVS bridge. If VM vNIC is SRIOV VF, then the mirror is applied to the hardware bridge. The interface name is specified for a physical interface, for example, GE0-0 or eth0.

Configuring SPAN Sessions

The SPAN session configuration has the following four parameters:

- Session number—Each SPAN session is identified with a unique number.
- Bridge name—The SPAN session is applied to a bridge. For VLAN mirroring, the bridge must be specified. The bridge name is optional if the source or destination interface is configured for the session.
- Source configuration—The source of the mirror traffic can be one of the following:
  - Packets entering (Rx), or exiting (Tx), or both. You can specify multiple interfaces of any type.
  - You can also specify all interfaces on the OVS bridge.
  - All packets entering a VLAN. You can also specify a list of VLANs.
- Destination configuration—The destination for the mirrored traffic can be one of the following:
  - The mirrored traffic can be sent to interfaces of any type.
  - The mirrored traffic can be sent to a speciifc VLAN. In this case, the original VLAN tag is stripped in the mirrored traffic in favor of the destination VLAN. This loss of original VLAN information might make the mirrored traffic hard to interpret.

To configure a SPAN session:

```
configure terminal
monitor session  2
bridge wan-br
source interface GE0-0
destination vm-vnic Linux2 0
commit
```

Verifying the SPAN Session Configuration

Use the **show system monitor session** command to verify the SPAN session configuration.

```
nfvis# show system monitor session
system monitor session 2
 bridge                wan-br
 destination_vlan      ""
 destination_interface vnic0
 source_vlans          ""
 source_rx_interfaces  "GE0-0"
 source_tx_interfaces  "GE0-0"
 source_all            false
 statistics            "tx_bytes=142660, tx_packets=1380"
```

Use the **show running-config monitor session** command to verify the interface configuration for a SPAN session:

```
nfvis# show running-config monitor session
monitor session 2
 destination vm-vnic Linux2 0
 source vm-vnic Linux1 0 both
 source interface GE0-0 both
```

SPAN Session APIs and Commands

APIs	Commands
• /api/config/monitor • /api/operational/monitor\?deep • /api/config/monitor\?deep • /api/operational/system/monitor/session\?deep	• monitor session • bridge • source • destination • show system monitor session • show monitor session status • show running-config monitor session

Configuration Examples for SPAN Session Scenarios

Example: SPAN Session Traffic on a Physical Interface

The following example shows how to configure all traffic coming in or going out on GE0-0 (physical interface) and VM Linux1 (vnic0). And traffic is mirrored to the VM Linux2 (vnic1). With this configuraton, any traffic arriving on vnet1 will be dropped.

Note An existing SPAN session will be in FAIL state after the system reboot. In this case, you need to recreate (delete and create) the SPAN session after the system bootup.

VM deployment interfaces:

- SPAN source: GE0-0 (traffic in both directions)
- SPAN source: Linux1/vnic0, and wan-net (traffic in both directions)
- • SPAN destination: Linux2/vnic0, and wan-net

```
nfvis# show running-config monitor session
monitor session 20
 destination vm-vnic Linux2 0
 source vm-vnic Linux1 0 both
 source interface GE0-0 both
!
nfvis#

nfvis# show system monitor session
system monitor session 20
 bridge                wan-br
 destination_vlan      ""
 destination_interface vnic11
 source_vlans          ""
 source_rx_interfaces  "vnic10, GE0-0"
 source_tx_interfaces  "vnic10, GE0-0"
 source_all            false
 statistics            "tx_bytes=142660, tx_packets=1380"
nfvis#

nfvis# show monitor session status
NUMBER  STATUS
------------------------
20      CREATE_SUCCESS
```

Example: SPAN Session Traffic on a LAN SRIOV

The following example shows how to configure all traffic coming in or going out on an SRIOV interface (VF0). It is also mirrored to VF1.

Note This scenario is applicable only to the Cisco ENCS.

VM deployment for VF-VF scenario:

CentOS_SRIOV, C3, and C5 are CentOS VMs with SRIOV support.

- CentOS_SRIOV: vnic0: wan-net/vnic1: LAN-SRIOV-1 (192.168.1.36)
- C3: vnic0: LAN-SRIOV3 (192.168.1.3)
- C5: vnic0: LAN-SRIOV5 (192.168.1.5)

SPAN destination and source:

- SPAN destination: CentOS_SRIOV (vnic0: wan-net/vnic1: LAN-SRIOV-1)
- SPAN source: C3 (vnic0: LAN-SRIOV-3); traffic in both directions (rx, tx)
- Ping target: C5 (vnic0: LAN-SRIOV-5)

```
nfvis# show running-config monitor session
monitor session 6
 destination vm-vnic CentOS_SRIOV 1
 source vm-vnic C3 0
!
nfvis#

nfvis# show system monitor session
system monitor session 6
 bridge                ""
 destination_vlan      ""
 destination_interface LAN-SRIOV-1
 source_vlans          ""
 source_rx_interfaces  LAN-SRIOV-3
 source_tx_interfaces  LAN-SRIOV-3
 source_all            ""
 statistics            ""
nfvis#

nfvis# show monitor session status
NUMBER  STATUS
------------------------
6       CREATE_SUCCESS
```

Example: SPAN Session Traffic on a VLAN

The following example shows how to configure the SPAN session for all traffic entering in VLAN 10 and 11. It is also mirrored to VLAN 20.

```
nfvis# show running-config monitor session
monitor session 11
 bridge lan-br
 destination vlan 20
 source vlan [ 10 11 ]
!

nfvis# show system monitor session
system monitor session 11
 bridge                lan-br
 destination_vlan      20
 destination_interface ""
 source_vlans          "10, 11"
 source_rx_interfaces  ""
 source_tx_interfaces  ""
 source_all            true
 statistics            "tx_bytes=0, tx_packets=0"

nfvis# show monitor session 11
NUMBER  STATUS
------------------------
11      CREATE_SUCCESS
```

Configuring Packet Capture

The Packet Capture feature helps you capture all packets being transmitted and received over physical and virtual network interface controllers (physical port and vNIC) for analysis. These packets are inspected to diagnose and solve network problems. Packets are stored in the*/data/intdatastore/pktcaptures* folder on the host server.

Benefits

- You can customize the configuration to capture specific packets such as Internet Control Message Protocol (ICMP), TCP, UDP, and Address Resolution Protocol (ARP).
- You can specify a time period over which packets are captured. The default is 60 seconds.

To configure packet capture on a physial port:

```
configure terminal
tcpdump port eth0

Output: pcap-location /data/intdatastore/pktcaptures/tcpdump_eth0.pcap
```

To configure packet capture on a vNIC:

```
configure terminal
tcpdump vnic tenant-name admin deployment-name 1489084431 vm-name ROUTER vnic-id 0 time 30

Output: pcap-location /data/intdatastore/pktcaptures/1489084431_ROUTER_vnic0.pcap
```

Types of Errors

Error	Scenario
Port/vnic not found	When non-existing interface is given as input.
File/directory not created	When the system is running out of disk space.
The **tcpdump** command fails	When the system is running out of disk space.

These errors are logged in the *nfvis_config.log*. By default, warnings and errors are logged,

Example: Debug Built-in Switch Issues

To monitor traffic problems related to built-in switch on an internal interface:

The regular traffic flow between int-LAN and GE1/0 is:

```
GE0-0-- vnic1---(VM)--vnic2--intLAN--GE1/0
```

The NFVIS portal has the capability to capture packets. In the network diagram, right click on any vertical line and a window pops up where you can select the duration of the capture. The packet capture starts on the selected interface link. At the end of the capture, a file is downloaded to your local machine. SPAN sessions are supported on both NFVIS host and the built-in switch.

The following is an example of SPAN in built-in switch:

1. From NFVIS system shell-access, get the password which can be used later.

```
cd /opt/switch-confd/
python decrypt_switch.py

<it will print out a string, it will be the password you need to use later>
8H7)gR348V4Byq4mwjiNt
```

2. From Cisco IMC complete the challenge-response authentication:

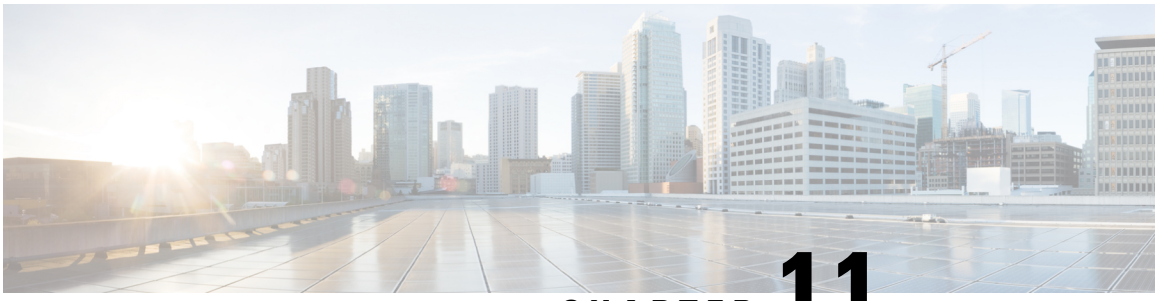
```
#connect debug-shell
#sldp
login <hit return>
it will print out the challenge string
enter the  respond string
# switch-con ge
user-name:cisco
password: <enter the sring we get from nfvis system shell>

User Name:cisco
Password:**********.   <this is the password you get from step 1 above>
```

3. To configure SPAN specify the source and distribution interface and direction of the packet flow. For example, if you want to mirror XG2 output packet to Ge0, connect an external packet capture tool in GE1/0 and you will see all packets flow from internal XG2. In the following example, the traffic between int_LAN and GE1/0 go through internal interface XE2 and traffic for XE2 interface is monitored:

```
nfvis(config)#monitor session 1 source interface XG 2 out
nfvis(config)#monitor session 1 destination interface GigabitEthernet 0
remember to unconfig it once you finish debuging.
nfvis(config)#no monitor session 1 destination
nfvis(config)#no monitor session 1 source interface XG 2
```

Packet Capture APIs and Commands

APIs	Commands
• /api/operations/packet-capture/tcpdump	• tcpdump port • tcpdump vnic



CHAPTER 11

Appendix

Event Notifications

nfvisEvent

Event Type	Notification Trigger	Notification Output Example
WAN_DHCP_RENEW	DHCP renew operation is performed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T18:06:46.142089+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Wan DHCP IP address is being renewed</status_message> <details>NA</details> <event_type>**WAN_DHCP_RENEW**</event_type> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
BRIDGE_DHCP_RENEW	Bridge DHCP renew operation is performed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2018-04-26T09:47:06.066264+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Bridge DHCP IP address is being renewed</status_message> <details>NA</details> <event_type>**BRIDGE_DHCP_RENEW**</event_type> </nfvisEvent> </notification>
INTF_STATUS_CHANGE	Interface status is changed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T18:12:09.963556+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <event_type>**INTF_STATUS_CHANGE**</event_type> <intf_name>eth7</intf_name> <intf_prv_op>up</intf_prv_op> <intf_op>down</intf_op> <intf_prv_link>down</intf_prv_link> <intf_link>down</intf_link> </nfvisEvent> </notification>
NETWORK_CREATE	A network is created.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-09-22T12:41:04.564298+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_message>Network created succesfully</status_message> <event_type>**NETWORK_CREATE**</event_type> <network_name>testn1</network_name> <network_bridge>test-net-br</network_bridge> <network_sriov>false</network_sriov> <network_vlan/> <network_trunk/> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
NETWORK_UPDATE	A network is updated.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-09-22T12:42:03.391986+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_message>Network updated succesfully</status_message> <event_type>**NETWORK_UPDATE**</event_type> <network_name>testn1</network_name> <network_bridge/> <network_sriov/> <network_vlan/> <network_trunk/> </nfvisEvent> </notification>
NETWORK_DELETE	A network is deleted.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-09-22T12:42:03.391986+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_message>Network deleted succesfully</status_message> <event_type>**NETWORK_DELETE**</event_type> <network_name>testn1</network_name> <network_bridge/> <network_sriov/> <network_vlan/> <network_trunk/> </nfvisEvent> </notification>
UPGRADE_REGISTER	System upgrade is registered.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T15:57:50.434636+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Upgrade package registration successful: Cisco_NFVIS_Upgrade-3.6.1-698-20170402_042811.nfvispkg</status_message> <event_type>**UPGRADE_REGISTER**</event_type> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
UPGRADE_APPLY	System upgrade is applied.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T16:02:43.885516+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Upgrade Process: In Progress</status_message> <event_type>**UPGRADE_APPLY**</event_type> </nfvisEvent> </notification>
ROTATED_LOGS_DELETE	Rotated logs older than 30 days are deleted by the system.	<?xml version="1.0" encoding="UTF-8"?> <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1"> <ok/> </rpc-reply> <?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T17:38:10.321152+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Deleted rotated logs from archive older than 30 days</status_message> <details>NA</details> <event_type>**ROTATED_LOGS_DELETE**</event_type> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
ROTATED_LOGS_DELETE	Older logs deleted by the system when the total file size of rotated logs exceeds 2GB.	<?xml version="1.0" encoding="UTF-8"?> <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1"> <ok/> </rpc-reply> <?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T17:42:10.321152+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Rotated logs had exceeded 2G, older logs have been deleted to make space</status_message> <details>NA</details> <event_type>**ROTATED_LOGS_DELETE**</event_type> </nfvisEvent> </notification>
REBOOT	system reboot	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2018-04-26T09:37:47.387525+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>System will be rebooted</status_message> <details>NA</details> <event_type>**REBOOT**</event_type> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
SHUTDOWN	system shutdown	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2018-04-26T09:47:06.066264+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>System will be shutdown</status_message> <details>NA</details> <event_type>**SHUTDOWN**</event_type> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
SECURE_OVERLAY_CREATING	create secure overlay	<notification <eventTime> 2018-11-02T04:23:02.641317+00:00 <nfvisEvent <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Secure Overlay mgmthub initial creation. Active local bridge: wan-br</status_message> <details>NA</details> <event_type>SECURE_OVERLAY_CREATING<event_type> <severity> INFO</severity> <hostname>nfvis</hostname> </nfvisEvent> </notification>
SECURE_OVERLAY_UP	Secure Overlay is UP	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2018-04-26T09:47:06.066264+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Secure Overlay mgmthub up. Active bridge: wan-br</status_message> <details>Secure overlay initial creation</details> <event_type>SECURE_OVERLAY_UP</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
WAN_DHCP_SWITCHOVER	WAN bridge toggle	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2018-04-26T09:47:06.066264+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Switch over to bridge wan2-br for auto DHCP enablement successful</status_message> <details>NA</details> <event_type>WAN_DHCP_SWITCHOVER</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
WAN_DHCP_TOGGLE_END	WAN bridge toggle	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2018-04-26T09:47:06.066264+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Disabling bridge toggle for auto DHCP enablement.</status_message> <details>NA</details> <event_type>WAN_DHCP_TOGGLE_END</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
ROUTE_DISTRIBUTION_START	To start route distribution	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Route Distribution initial creation. Neighbor Address: 172.25.221.106</status_message> <details>NA</details> <event_type>ROUTE_DISTRIBUTION_START</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
ROUTE_DISTRIBUTION_DOWN	Route distribution is down	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Neighbor Address: 172.25.221.106</status_message> <details>NA</details> <event_type>ROUTE_DISTRIBUTION_DOWN</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
ROUTE_DISTRIBUTION_ERROR	Route distribution in error	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Neighbor Address: 172.25.221.106</status_message> <details>NA</details> <event_type>ROUTE_DISTRIBUTION_ERROR</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
ROUTE_DISTRIBUTION_DELETE	Route distribution deleted	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>All Neighbor Addresses deleted</status_message> <details>NA</details> <event_type>ROUTE_DISTRIBUTION_DELETE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
ROUTE_DISTRIBUTION_UP	Route distribution up	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Neighbor Address: 172.25.221.106</status_message> <details>NA</details> <event_type>ROUTE_DISTRIBUTION_UP</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
OVS_DPDK_SUCCESS	Enable DPDK	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>OVS-DPDK enabled</status_message> <details>NA</details> <event_type>OVS_DPDK_SUCCESS</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
OVS_DPDK_FAILURE	DPDK failure	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Unable to allocate CPU</status_message> <details>NA</details> <event_type>OVS_DPDK_FAILURE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
PNIC_SRIOV_ENABLE	Enable SR-IOV to 2 vfs	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Number of VFs 2</status_message> <details>NA</details> <event_type>PNIC_SRIOV_ENABLE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
PNIC_SRIOV_DISABLE	Disable SR-IOV	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Number of VFs 0</status_message> <details>NA</details> <event_type>PNIC_SRIOV_DISABLE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
PNIC_SRIOV_ENABLE	Enable of disable SR-IOV in error	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Number of VFs 2</status_message> <details>NA</details> <event_type>PNIC_SRIOV_ENABLE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
PNIC_SRIOV_UPDATE	Set switchmode to veb/vepa	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Switchmode vepa</status_message> <details>NA</details> <event_type>PNIC_SRIOV_UPDATE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
PNIC_SRIOV_UPDATE	Set switchmode to veb/vepa in error	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Switchmode vepa</status_message> <details>NA</details> <event_type>PNIC_SRIOV_UPDATE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
PROMISC_ENABLED	Enable promiscuous mode	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2020-03-27T19:58:52.333682+00:00</eventTime> <nfvisEvent <user_id>admin</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>GE0-0: promiscuous mode enabled</status_message> <details>GE0-0: promiscuous mode enabled</details> <event_type>PROMISC_ENABLED</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

vmlcEvent

Event Type	Notification Trigger	Notification Output Example
CREATE_IMAGE	The VM image is registered.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:12:30.76+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Image creation completed successfully.</status_message> <image>isrv-universalk9.16.03.01.tar.gz</image> <vm_source></vm_source> <vm_target></vm_target> <event> <type>**CREATE_IMAGE**</type> </event> </vmlcEvent> </notification>
DELETE_IMAGE	The VM image is unregistered.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:14:51.169+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Image deletion completed successfully.</status_message> <image>isrv-universalk9.16.03.01.tar.gz</image> <vm_source></vm_source> <vm_target></vm_target> <event> <type>**DELETE_IMAGE**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
CREATE_FLAVOR	A flavor is created.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:12:29.685+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Flavor creation completed successfully.</status_message> <flavor>ISRv-small</flavor> <vm_source></vm_source> <vm_target></vm_target> <event> <type>**CREATE_FLAVOR**</type> </event> </vmlcEvent> </notification>
DELETE_FLAVOR	A flavor is deleted.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:14:51.425+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Flavor deletion completed successfully.</status_message> <flavor>ISRv-small</flavor> <vm_source></vm_source> <vm_target></vm_target> <event> <type>**DELETE_FLAVOR**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_DEPLOYED	The VM is deployed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:19:16.927+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>VIM Driver: VM successfully created, VM Name: [SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c]</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> <interfaces> <interface> <nicid>0</nicid> <port_id>vnet0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.2</ip_address> <mac_address>52:54:00:31:c5:7f</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <port_id>vnet1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:59:52:41</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> </interfaces> </vm_source> <vm_target></vm_target> <event> <type>**VM_DEPLOYED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_ALIVE	The state of a monitored VM becomes ACTIVE.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:22:47.306+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>VM_Alive event received, VM ID: [SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c]</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> <interfaces> <interface> <nicid>0</nicid> <port_id>vnet0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.2</ip_address> <mac_address>52:54:00:31:c5:7f</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <port_id>vnet1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:59:52:41</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> </interfaces> </vm_source> <vm_target></vm_target> <event> <type>**VM_ALIVE**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_UNDEPLOYED	The VM is undeployed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:31:40.6+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>204</status_code> <status_message>VIM Driver: VM successfully deleted</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> <interfaces> <interface> <nicid>0</nicid> <port_id>vnet0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.2</ip_address> <mac_address>52:54:00:31:c5:7f</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <port_id>vnet1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:59:52:41</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> </interfaces> </vm_source> <vm_target></vm_target> <event> <type>**VM_UNDEPLOYED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
SERVICE_UPDATED	The VM is updated.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:51:45.5+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Service group update completed successfully</status_message> <depname>1479342258</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <depid>827e871a-30d5-4f5f-a05a-263b7ee3a734</depid> <vm_source></vm_source> <vm_target></vm_target> <event> <type>**SERVICE_UPDATED**</type> </event> </vmlcEvent> </notification>
VM_STOPPED	The VM is stopped per VM action requrest.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:26:05.762+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Successfully stopped VM [SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c].</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_STOPPED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_STARTED	The VM is started per VM action request.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:26:40.398+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Started VM [SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c].</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_STARTED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_REBOOTED	The VM is rebooted per VM action request.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:36:56.5+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Rebooted VM [SystemAdminTena_ROUTER_0_f17fc494-8535-4b05-b88d-f0fd2effdc7d]</status_message> <depname>1479342258</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>827e871a-30d5-4f5f-a05a-263b7ee3a734</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d918a3b1-f2a9-4065-9d8e-2135b0a37d87</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_REBOOTED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_RECOVERY_INIT	A monitored VM is not reachable.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T16:27:51.627+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Recovery event for VM [SystemAdminTena_ROUTER_0_40ae18be-5930-4d94-95ff-dbb0b56ef12b] trigge red. Processing Auto healing. Proceeding with Recovery.</status_message> <depname>1479328919</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>9e7fe4f8-a5f4-4a6d-aad7-121405be4ba4</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>000883fc-77f3-4b9e-aaf6-0f31d88a8f67</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_RECOVERY_INIT**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_RECOVERY_REBOOT	Recovery reboot starts for the monitored VM, which is not reachable.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T16:27:53.979+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>VM [SystemAdminTena_ROUTER_0_40ae18be-5930-4d94-95ff-dbb0b56ef12b] is being rebooted. </statu s_message> <depname>1479328919</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>9e7fe4f8-a5f4-4a6d-aad7-121405be4ba4</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>000883fc-77f3-4b9e-aaf6-0f31d88a8f67</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_RECOVERY_REBOOT**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_RECOVERY_COMPLETE	Recovery reboot completes for the monitored VM, which is not reachable.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T16:31:26.934+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Successfully recovered VM [SystemAdminTena_ROUTER_0_40ae18be-5930-4d94-95ff-dbb0b56ef12b]. status_message> <depname>1479328919</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>9e7fe4f8-a5f4-4a6d-aad7-121405be4ba4</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>000883fc-77f3-4b9e-aaf6-0f31d88a8f67</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target> <vmid>000883fc-77f3-4b9e-aaf6-0f31d88a8f67</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> <interfaces> <interface> <nicid>0</nicid> <port_id>vnet0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.2</ip_address> <mac_address>52:54:00:7b:3f:de</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <port_id>vnet1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:96:8a:4d</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> </interfaces> </vm_target> <event> <type>**VM_RECOVERY_COMPLETE**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_MONITOR_UNSET	Monitoring is disabled per VM action request.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-18T13:36:43.613+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Unset monitor completed successfully</status_message> <depname>1479413090</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>742dd335-330c-4bf0-a75d-a44003c645c5</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>23ec3793-37ab-4ec2-a978-a10e08585fdd</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_MONITOR_UNSET**</type> </event> </vmlcEvent> </notification>
VM_MONITOR_SET	Monitoring is enabled per VM action request.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-18T13:40:15.276+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Set monitor completed successfully</status_message> <depname>1479413090</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>742dd335-330c-4bf0-a75d-a44003c645c5</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>23ec3793-37ab-4ec2-a978-a10e08585fdd</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_MONITOR_SET**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_UPDATED	VM's flavor is changed.	

Event Type	Notification Trigger	Notification Output Example
		<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-12-08T00:50:39.034+00:00</eventTime> <vmlcEvent xmlns='http://www.cisco.com/nfvis/vm_lifecycle'> <status>SUCCESS</status> <status_code>200</status_code> <status_message>VM is resized with flavor [ISRv-medium].</status_message> <user_name>admin</user_name> <depname>1512766000</depname> <tenant>admin</tenant> <tenant_id>adminUUID</tenant_id> <depid>92c11aa1-f6dd-47d1-948f-c8c65b9ef70f</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>1a6f587e-2779-4087-b84d-c0a2c8a481b1</vmid> <vmname>1512766000_ROUTER_0_60d15064-0c6d-49b9-aa4a-80587f626004</vmname> <hostid>NFVIS</hostid> <hostname>nfvis</hostname> <interfaces> <interface> <nicid>0</nicid> <type>virtual</type> <port_id>vnic0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.3</ip_address> <mac_address>52:54:00:3c:ee:5b</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <type>virtual</type> <port_id>vnic1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:70:06:4a</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> <interface> <nicid>2</nicid> <type>virtual</type> <port_id>vnic2</port_id> <network>lan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:c7:30:1c</mac_address> <netmask>255.255.255.0</netmask> <gateway>192.168.1.1</gateway> </interface> </interfaces> </vm_source> <event> <type>VM_UPDATED</type> </event>

Event Type	Notification Trigger	Notification Output Example
		</vmlcEvent></notification>]]>]]> <?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-12-08T00:50:39.06+00:00</eventTime> <vmlcEvent xmlns='http://www.cisco.com/nfvis/vm_lifecycle'> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Service group update completed successfully</status_message> <user_name>admin</user_name> <depname>1512766000</depname> <tenant>admin</tenant> <tenant_id>adminUUID</tenant_id> <depid>92c11aa1-f6dd-47d1-948f-c8c65b9ef70f</depid> <event> <type>SERVICE_UPDATED</type> </event> </vmlcEvent>

Event Type	Notification Trigger	Notification Output Example
VM_UPDATED	VNIC is added, deleted or updated.	

Event Type	Notification Trigger	Notification Output Example

```
<?xml version="1.0" encoding="UTF-8"?>
<notification
 xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
 <eventTime>2017-12-08T02:10:56.184+00:00</eventTime>
 <vmlcEvent
  xmlns='http://www.cisco.com/nfvis/vm_lifecycle'>
  <status>SUCCESS</status>
  <status_code>200</status_code>
  <status_message>Added 1 interface: [managed, net=my-net-1, nicid=3]
Updated 2 interface: [managed, net=lan-net, nicid=1],[managed,
net=wan-net, nicid=2]</status_message>
  <user_name>admin</user_name>
  <depname>1512766000</depname>
  <tenant>admin</tenant>
  <tenant_id>adminUUID</tenant_id>
  <depid>92c11aa1-f6dd-47d1-948f-c8c65b9ef70f</depid>
  <vm_group>ROUTER</vm_group>
  <vm_source>
   <vmid>1a6f587e-2779-4087-b84d-c0a2c8a481b1</vmid>

<vmname>1512766000_ROUTER_0_60d15064-0c6d-49b9-aa4a-80587f626004</vmname>

   <hostid>NFVIS</hostid>
   <hostname>nfvis</hostname>
   <interfaces>
    <interface>
     <nicid>0</nicid>
     <type>virtual</type>
     <port_id>vnic0</port_id>
     <network>int-mgmt-net</network>
     <subnet>N/A</subnet>
     <ip_address>10.20.0.3</ip_address>
     <mac_address>52:54:00:3c:ee:5b</mac_address>
     <netmask>255.255.255.0</netmask>
     <gateway>10.20.0.1</gateway>
    </interface>
    <interface>
     <nicid>1</nicid>
     <type>virtual</type>
     <port_id>vnic1</port_id>
     <network>lan-net</network>
     <subnet>N/A</subnet>
     <mac_address>52:54:00:70:06:4a</mac_address>
     <netmask>255.255.255.0</netmask>
     <gateway>192.168.1.1</gateway>
    </interface>
    <interface>
     <nicid>2</nicid>
     <type>virtual</type>
     <port_id>vnic2</port_id>
     <network>wan-net</network>
     <subnet>N/A</subnet>
     <mac_address>52:54:00:c7:30:1c</mac_address>
     <netmask>255.255.255.0</netmask>
     <gateway>172.19.181.152</gateway>
    </interface>
    <interface>
     <nicid>3</nicid>
     <type>virtual</type>
     <port_id>vnic3</port_id>
```

Event Type	Notification Trigger	Notification Output Example
		<network>my-net-1</network> <subnet>N/A</subnet> <mac_address>52:54:00:66:b5:c1</mac_address> </interface> </interfaces> </vm_source> <event> <type>VM_UPDATED</type> </event> </vmlcEvent>

Syslog Messages

Event	Trigger Condition	Syslog Messages
NETWORK_CREATE	create a new network	nfvis %SYS-6-NETWORK_CREATE: Network my-net created successfully
NETWORK_UPDATE	modify an existing network	nfvis %SYS-6-NETWORK_UPDATE: Network my-net updated successfully
NETWORK_DELETE	delete a network	nfvis %SYS-6-NETWORK_DELETE: Network my-net deleted successfully
BRIDGE_CREATE	create a new bridge	nfvis %SYS-6-BRIDGE_CREATE: Bridge created successfully: my-bridge
BRIDGE_UPDATE	modify an existing bridge	nfvis %SYS-6-BRIDGE_UPDATE: Updated bridge successfully: my-bridge
BRIDGE_DELETE	delete a bridge	nfvis %SYS-6-BRIDGE_DELETE: Bridge deleted successfully: my-bridge
WAN_DHCP_RENEW	dhcp renew from wan interface	nfvis %SYS-6-WAN_DHCP_RENEW: wan-br DHCP IP address is being renewed
BRIDGE_DHCP_RENEW	bridge dhcp renew	nfvis %SYS-6-BRIDGE_DHCP_RENEW: Bridge DHCP IP address is being renewed
MGMT_DHCP_RENEW	dhcp renew from MGMT interface	nfvis %SYS-6-MGMT_DHCP_RENEW: wan-br DHCP IP address is being renewed
INTF_STATUS_CHANGE	interface status change	nfvis %SYS-6-INTF_STATUS_CHANGE: Interface eth0, changed state to up
UPGRADE_REGISTER	upgrade package registration	nfvis %SYS-6-UPGRADE_REGISTER: Upgrade package registration successful: Cisco_NFVIS_Upgrade-3.6.1-698-20170402_042811.nfvispkg
UPGRADE_APPLY	upgrade process	nfvis %SYS-6-UPGRADE_APPLY: Upgrade Process: In Progress

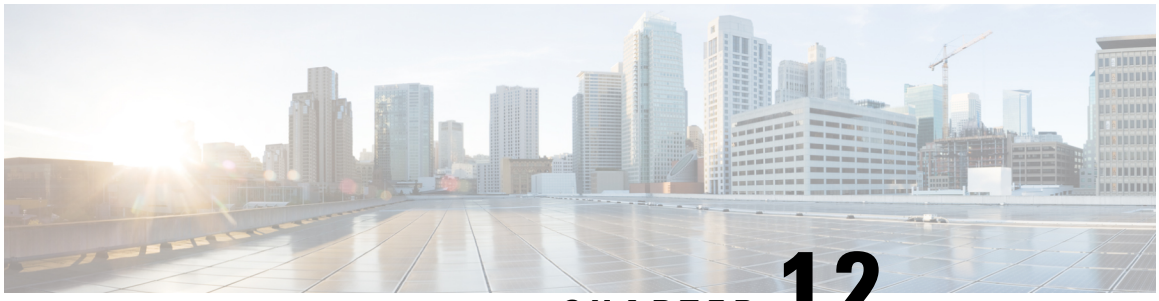
Event	Trigger Condition	Syslog Messages
RBAC_USER_CREATE	create a new user	nfvis %SYS-6-RBAC_USER_CREATE: Created user admin as administrators successfully
RBAC_USER_PASSWORD_UPDATE	change user's password	nfvis %SYS-6-RBAC_USER_PASSWORD_UPDATE: Set admin password successfully
RBAC_USER_ROLE_UPDATE	change user's role	nfvis %SYS-6-RBAC_USER_ROLE_UPDATE: Modified user: somebody successfully
RBAC_USER_DELETE	delete a user	nfvis %SYS-6-RBAC_USER_DELETE: Deleted rbac user successfully: somebody
RBAC_USERS_INACTIVATED	disable the user	nfvis %SYS-6-RBAC_USERS_INACTIVATED: Following users have been marked as INACTIVE, [user1, user2]. Please take necessary action.
RBAC_USER_ACTIVATED	activate the user	nfvis %SYS-6-RBAC_USER_ACTIVATED: Modified user user1 successfully.
RBAC_PWD_EXPIRED	password expired	nfvis %SYS-6-RBAC_PWD_EXPIRED: User user1's password is older than 60 days. Please reset password.
RBAC_LOGIN_FAILURE	invalid user login	nfvis %SYS-3-RBAC_LOGIN_FAILURE: Login with invalid username from maapi failed
SECURITY_SERVER_CREATE	create server config	nfvis %SYS-6-SECURITY_SERVER_CREATE: TACACS+ server config created successfully.
SECURITY_SERVER_UPDATE	update server config	nfvis %SYS-6-SECURITY_SERVER_UPDATE: TACACS+ server configuration updated successfully.
SECURITY_SERVER_DELETE	delete server config	nfvis %SYS-6-SECURITY_SERVER_DELETE: TACACS+ server deleted successfully.
AAA_TYPE_CREATE	create AAA authentication type	nfvis %SYS-6-AAA_TYPE_CREATE: AAA authentication type TACACS created successfully.
AAA_TYPE_UPDATE	update AAA authentication type	nfvis %SYS-6-AAA_TYPE_UPDATE: AAA authentication type TACACS+ updated successfully. AAA authentication updated to use TACACS+ server
RECREATE_CERTIFICATE	recreate self-sign certificate	nfvis %SYS-6-RECREATE_CERTIFICATE: Self Signed Certificate re-created. Application connection may become temporarily unavailable.
CERT_CSR_CREATE	create a CSR file	nfvis %SYS-6-CERT_CSR_CREATE: signing-request created /data/intdatastore/download/nfvis.csr
CERT_SWITCH_CERT	switch to use different certificate	nfvis %SYS-6-CERT_SWITCH_CERT: switch certificate from ca-signed to self-signed.
CERT_CA_CERT_INSTALL	install CA signed certificate	nfvis %SYS-6-CERT_CA_CERT_INSTALL: ca-signed certificate file:// installed

Event	Trigger Condition	Syslog Messages
REBOOT	system reboot	nfvis %SYS-6-REBOOT: System will be rebooted
SHUTDOWN	system shutdown	nfvis %SYS-6-SHUTDOWN: System will be shutdown
LOGGING_FAILURE	logging failure	nfvis %SYS-6-LOGGING_FAILURE: Unable to write to log file nfvis_config.log. Log message: log_config.CONFIG_LOGGER: File not found.
DISK_SPACE_ALMOST_FULL	disk space almost full	nfvis %SYS-6-DISK_SPACE_ALMOST_FULL: 'lv_data' currently occupies 95% of available disk space, which is more than or equal to the threshold of 90%.
ROTATED_LOGS_DELETE	delete rotated logfiles when accumulated rotated log files reach 2GB	nfvis %SYS-6-ROTATED_LOGS_DELETE: Deleted rotated logs from archive older than 30 days
TIME_UPDATE	Change system time manually	nfvis %SYS-6-TIME_UPDATE: Manual time updated successfully Manual time is now set to 2018-04-26 11:43:00
TIMEZONE_UPDATE	Change system timezone	nfvis %SYS-6-TIMEZONE_UPDATE: Timezone updated successfully. Timezone is now set to US/Eastern
FILE_COPY_STATUS	copy status of file	nfvis %SYS-6-FILE_COPY_STATUS: hostaction.py Copied Successfully.
CREATE_IMAGE	create image	nfvis %SYS-6-CREATE_IMAGE: Image creation successful: TinyLinux.tar.gz
DELETE_IMAGE	delete image	nfvis %SYS-6-DELETE_IMAGE: Image deletion successful: TinyLinux.tar.gz
CREATE_FLAVOR	create flavor	nfvis %SYS-6-CREATE_FLAVOR: Profile creation successful: small
DELETE_FLAVOR	delete flavor	nfvis %SYS-6-DELETE_FLAVOR: Profile deletion successful: small
VM_DEPLOYED	vm deployment	nfvis %SYS-6-VM_DEPLOYED: VM deployment successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_ALIVE	vm alive	nfvis %SYS-6-VM_ALIVE: VM active successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
SERVICE_ALIVE	service alive	nfvis %SYS-6-SERVICE_ALIVE: Service group deployment completed successfully!

Event	Trigger Condition	Syslog Messages
VM_UNDEPLOYED	vm undeployed	nfvis %SYS-6-VM_UNDEPLOYED: VM undeployment successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c SERVICE_UNDEPLOYED service undeployed nfvis %SYS-6-SERVICE_UNDEPLOYED: Service group undeployment completed successfully
VM_UPDATED (update flavor)	vm updated	nfvis %SYS-6-VM_UPDATED: VM update successful: VM is resized with flavor [ISRv-medium].
VM_UPDATED (vnic add / delete / update)	vm updated	nfvis %SYS-6-VM_UPDATED: VM update successful: Added 1 interface: [managed, net=my-net-1, nicid=3] Updated 2 interface: [managed, net=lan-net, nicid=1],[managed, net=wan-net, nicid=2]
SERVICE_UPDATED	service updated	nfvis %SYS-6-SERVICE_UPDATED: Service group update completed successfully
VM_STOPPED	vm stopped	nfvis %SYS-6-VM_STOPPED: VM stop successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_STARTED	vm started	nfvis %SYS-6-VM_STARTED: VM start successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_REBOOTED	vm rebooted	nfvis %SYS-6-VM_REBOOTED: VM reboot successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_RECOVERY_INIT	vm recovery initiation	nfvis %SYS-6-VM_RECOVERY_INIT: VM recovery initiation successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_RECOVERY_REBOOT	vm recovery reboot	nfvis %SYS-6-VM_RECOVERY_REBOOT: VM recovery reboot successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_RECOVERY_COMPLETE	vm recovery complete	nfvis %SYS-6-VM_RECOVERY_COMPLETE: VM recovery successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_MONITOR_UNSET	vm monitoring unset	nfvis %SYS-6-VM_MONITOR_UNSET: Unsetting VM monitoring successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_MONITOR_SET	vm monitoring set	nfvis %SYS-6-VM_MONITOR_SET: Setting VM monitoring successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
ROTATED_LOGS_DELETE (When logs older than 30 days are present)	delete rotated logs	nfvis %SYS-6-ROTATED_LOGS_DELETE: Deleted rotated logs from archive older than 30 days

Event	Trigger Condition	Syslog Messages
ROTATED_LOGS_DELETE (When Log file size exceed 2GB, older logs are deleted)	delete rotated logs	nfvis %SYS-6-ROTATED_LOGS_DELETE: Rotated logs had exceeded 2G, older logs have been deleted to make space
CIMC_PASSWORD_UPDATE	cimc password update operation	nfvis %SYS-6-CIMC_PASSWORD_UPDATE: CIMC password change is successful
BIOS_PASSWORD_UPDATE	bios password update operation	nfvis %SYS-6-BIOS_PASSWORD_UPDATE: BIOS password change is successful
SECURE_OVERLAY_CREATING	create secure overlay	nfvis %SYS-6-SECURE_OVERLAY_CREATING: Secure Overlay mgmthub initial creation. Active local bridge: wan-br
SECURE_OVERLAY_UP	secure overlay is up	nfvis %SYS-6-SECURE_OVERLAY_UP: Secure Overlay mgmthub up. Active bridge: wan-br Secure Overlay up after network interuption
SECURE_OVERLAY_DELETE	secure overlay is deleted	nfvis %SYS-6-SECURE_OVERLAY_DELETE: Secure Overlay deleted
SECURE_OVERLAY_ERROR	error in secure overlay	nfvis %SYS-3-SECURE_OVERLAY_ERROR: Secure Overlay mgmthub creation in error. Active bridge: wan-br Secure overlay initial creation nfvis %SYS-3-SECURE_OVERLAY_ERROR: Secure Overlay mgmthub creation in error. Active bridge: wan-br Cannot ping remote system ip address 10.0.0.1
WAN_DHCP_SWITCHOVER	WAN bridge toggle	nfvis %SYS-6-WAN_DHCP_SWITCHOVER: Switch over to bridge wan-br for auto DHCP enablement successful
WAN_DHCP_TOGGLE_END	WAN bridge toggle	nfvis %SYS-6-WAN_DHCP_TOGGLE_END: Disabling bridge toggle for auto DHCP enablement.
ROUTE_DISTRIBUTION_DOWN	Route distribution down	nfvis %SYS-6-ROUTE_DISTRIBUTION_DOWN: Neighbor Address: 172.25.221.106
ROUTE_DISTRIBUTION_START	Route distribution start	nfvis %SYS-6-ROUTE_DISTRIBUTION_START: Route Distribution initial creation. Neighbor Address: 172.25.221.106
ROUTE_DISTRIBUTION_ERROR	Route distribution in error state	nfvis %SYS-3-ROUTE_DISTRIBUTION_ERROR: Neighbor Address: 172.25.221.106
ROUTE_DISTRIBUTION_DELETE	Route distribution deleted	nfvis %SYS-6-ROUTE_DISTRIBUTION_DELETE: All Neighbor Addresses deleted
ROUTE_DISTRIBUTION_UP	Route distribution up	nfvis %SYS-3-ROUTE_DISTRIBUTION_UP: Neighbor Address: 172.25.221.106

Event	Trigger Condition	Syslog Messages
OVS_DPDK_SUCCESS	Enable DPDK	nfvis %SYS-3-OVS_DPDK_SUCCESS: OVS-DPDK enabled
OVS_DPDK_FAILURE	DPDK failure	nfvis %SYS-3-OVS_DPDK_FAILURE: Unable to allocate CP
BACKUP_INIT	Backup configuration initiation	nfvis %SYS-6-BACKUP_INIT: Starting backup: configuration-xxx
BACKUP_SUCCESS	Backup configuration successful	nfvis %SYS-6-BACKUP_SUCCESS: Backup configuration-xxx completed successfully
BACKUP_FAILURE	Backup configuration failure	nfvis %SYS-3-BACKUP_FAILURE: Backup configuration-xxx failed
RESTORE_INIT	Restore initiation	nfvis %SYS-6-RESTORE_INIT: Restore started
RESTORE_ SUCCESS	Successful restore	nfvis %SYS-6-RESTORE_ SUCCESS: Restore successful
RESTORE_FAILURE	Failure to restore	nfvis %SYS-3-RESTORE_FAILURE: Restore failed - internal error



CHAPTER 12

Glossary

Terms	Description
BIOS	BIOS is firmware used to perform hardware initialization during the booting process, and to provide runtime services for operating systems and programs. The BIOS firmware comes pre-installed on a personal computer's system board, and it is the first software to run when powered on.
Cisco IMC	The Cisco Integrated Management Controller (IMC) is the management service for the C-Series servers. CIMC runs within the server. You can use a web-based GUI or SSH-based CLI to access, configure, administer, and monitor the server.
DPDK	The Data Plane Development Kit (DPDK) is a set of data plane libraries and network interface controller drivers for fast packet processing.
DTLS	Datagram Transport Layer Security (DTLS) is a communications protocol that provides security for datagram-based applications by allowing them to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery.
IPSec	Internet Protocol Security (IPsec) is a secure network protocol suite that authenticates and encrypts the packets of data to provide secure encrypted communication between two computers over an Internet Protocol network
LACP	Link Aggregation Control Protocol (LACP) is a protocol for the collective handling of multiple physical ports that can be seen as a single channel for network traffic purposes.
LLDP	Link Layer Discovery Protocol (LLDP) is a vendor independent link layer protocol used by network devices for advertising their identity, capabilities to neighbors on a LAN segment.
MIB	Management Information Base (MIB) is a database of the objects that can be managed on a device. The managed objects, or variables, can be set or read to provide information on the network devices and interfaces.
NETCONF	A Network Configuration Protocol (NETCONF) is a protocol defined by the IETF to install, edit, and delete the configuration of network devices.
NGIO	Next Generation Input/Output (NGIO)

Terms	Description
PnP	Plug and Play (PnP) increases speed and reduces complexity of device deployments.
Port Channels	Port channels combine individual links into a group to create a single logical link that provides the aggregate bandwidth of up to eight physical links.
RADIUS	Remote Authentication Dial-In User Service (RADIUS) is a networking protocol, operating on port 1812 that provides centralized Authentication, Authorization, and Accounting management for users who connect and use a network service.
REST API	Representational state transfer (REST) suggests to create an object of the data requested by the client and send the values of the object in response to the user.
Service Chaining	Service chaining allows data traffic to be rerouted through one or more services, such as firewall, load balancer, and intrusion detection and prevention (IDP) devices.
SNMP	Simple Network Management Protocol (SNMP) is a framework used for managing devices on the internet. It provides a set of operations for monitoring and managing the internet.
SPAN	Switched Port Analyzer (SPAN) feature, which is sometimes called port mirroring or port monitoring, selects network traffic for analysis by a network analyzer.
Spanning Tree	Spanning Tree Protocol (STP) is a Layer 2 protocol that runs on bridges and switches. The main purpose of STP is to ensure that you do not create loops when you have redundant paths in your network.
SR-IOV	Single Root I/O Virtualization (SR-IOV) allows multipleVMs running a variety of guest operating systems to share a single PCIe network adapter within a host server. SR-IOV allows a VM to move data directly to and from the network adapter, bypassing the hypervisor for increased network throughput and lower server CPU burden.
TACACS	Terminal Access Controller Access-Control System (TACACS) refers to a family of related protocols handling remote authentication and related services for networked access control through a centralized server.
UEFI	The Unified Extensible Firmware Interface (UEFI) is a specification that defines a software interface between an operating system and platform firmware.
virtio	Virtual input/output (virtio) is a virtualization standard for network and disk device drivers where just the guest's device driver "knows" it is running in a virtual environment, and cooperates with the hypervisor.
VM	A virtual machine (VM) is an emulation of a computer system. Virtual machines are based on computer architectures and provide functionality of a physical computer. Their implementations may involve specialized hardware, software, or a combination.
VNF	Virtual Network Functions (VNFs), the software version of network appliances such as a router, firewall, load-balancer etc