

Cisco Enterprise Network Function Virtualization Infrastructure Software Configuration Guide, Release 3.10.x

First Published: 2018-12-05

Last Modified: 2019-02-15

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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2017–2018 Cisco Systems, Inc. All rights reserved.



CONTENTS

Full Cisco Trademarks with Software License ?

PREFACE

About this Document xi

About this Document xi

Audience xii

Related Documentation xii

Communications, Services, and Additional Information xii

PART I

NFVIS 15

CHAPTER 1

About Cisco Enterprise NFVIS 1

Benefits of Cisco Enterprise NFVIS 1

Supported Hardware Platforms 2

Supported VMs 3

Key Tasks You can Perform Using Cisco Enterprise NFVIS 4

CHAPTER 2

Installing Cisco Enterprise NFVIS Using the KVM Console 5

Installation Prerequisites 5

Image Signing and Verification 6

RPM Signing 6

RPM Signature Verification 6

Image Integrity Verification Using sha256sum 6

Entering BIOS Setup 7

Installing Cisco Enterprise NFVIS on the Cisco UCS C220 M4 and M5 Rack Servers or Cisco CSP Devices 7

Logging Into the CIMC GUI 7

Installing Cisco Enterprise NFVIS on Cisco UCS E-Series Servers 8

Sample Configuration on the Cisco ISR Router to Bring Up a Cisco UCS E Server 10
Installing Cisco Enterprise NFVIS on a Cisco ENCS 5100 and 5400 12
Install Cisco Enterprise NFVIS on Cisco ENCS 5104 14

CHAPTER 3 **Setting Up System Configuration** 17
Default System Configuration on the Cisco ENCS 17
Default System Configuration on the Cisco UCS C220 M4 Server and Cisco CSP 2100 19
Default System Configuration on the Cisco UCS E-Series Servers 20
Accessing NFVIS 20
Configuring VLAN for NFVIS Management Traffic 26
Configuring System Routes 26
User Roles and Authentication 27
Rules for User Passwords 27
Creating Users and Assigning Roles 27
Configuring Minimum Length for Passwords 28
Configuring Password Lifetime 29
Deactivating Inactive User Accounts 29
Activating an Inactive User Account 29
Certification 30
Secure Copy Command 30
Configuring the IP Receive ACL 31
Port 22222 and Management Interface ACL 32
Configuring Your Banner and Message of the Day 32
Setting the System Time Manually or With NTP 33
Enable or Disable Access to NFVIS Portal 34
Configuring System Logs 35
Network File System Support 36
Secure Boot of host 37
CIMC Access Control 38
CIMC Access using NFVIS 38
BIOS-CIMC Update 39
BIOS and CIMC Password 39
NFVIS Password Recovery 40
UEFI Secure Boot on ENCS 5400 41

DPDK Support on NFVIS 42
DPDK Support for NFVIS 3.10.x 42
DPDK VM Migration for NFVIS 3.11.x 43
Enhancements to DPDK Support in NFVIS 3.12.x 43
Import and Export NFVIS VM 45
Backup and Restore NFVIS and VM Configurations 48
Dynamic SR-IOV 57

CHAPTER 4 Cisco Network Plug-n-Play Support 61
PnP Discovery Methods 62
Configuring PnP Discovery Methods 63
PnP Action 66

CHAPTER 5 Overview of VM Life Cycle Management 67
Workflow of VM Life Cycle Management 67
Uploading VM Images to an NFVIS Server 69
VM Bootstrap Configuration Options with a VM Deployment 70
OpenStack Configuration Drive Support for Third Party VMs 71
Performing Resource Verification 72
Configuring Management IP Subnet 73
VM States 73

CHAPTER 6 VM Deployment Scenarios 75
Registering VM Images 75
Single VM Deployment 76
Steps for Deploying a VM 76
Service Chaining of VMs 79
Service Chaining with two VM Images 79
Steps for Service Chaining with Two VM Images 79
Service Chaining of Multiple VMs with Windows or Linux Servers 80
Steps for Service Chaining of Multiple VMs with Windows or Linux Servers 80

CHAPTER 7 SPAN Session or Port Mirroring 81
About SPAN Sessions 81

Configuring SPAN Sessions 81
Configuration Examples for SPAN Session Scenarios 83
Example: SPAN Session Traffic on a Physical Interface 83
Example: SPAN Session Traffic on a LAN SRIOV 84
Example: SPAN Session Traffic on a VLAN 85

CHAPTER 8 **Configuring Packet Capture** 87

CHAPTER 9 **VM Image Packaging** 89
VM Image Packaging Utility 89
Contents 89
Usage 90
NFVIS Specific Enhancements 96
VM Packaging Utility Usage Examples 96
Standard VM Image Packaging 98
Generating a VM Package 98
Appendix 98
VM Image Package Files 98
Package Manifest File 99
Bootstrap Configuration File 100
VM Image Properties File 100
Example: Package.mf 105
Example: Image Properties 105
Example: Bootstrap Configuration File 106
Image Properties Template File 107

CHAPTER 10 **Upgrading Cisco Enterprise NFVIS** 109

CHAPTER 11 **Configuring vBranch High Availability** 113
Prerequisites for vBranch HA 113
vBranch HA Design and Topology 114
Enable Virtual NIC Failure Detection with Track Feature 114
Isolating LAN and Transit Link Traffic for vBranch HA 116
Packet Flow for vBranch HA 118

Configuration Examples for vBranch HA 119
Example: Active Cisco ENCS Configuration with ISRv1 119
Example: Standby Cisco ENCS Configuration with ISRv2 121
Cisco ENCS Failure Points 122

CHAPTER 12 Cisco ENCS Single WAN IP Deployment Scenarios 127
Single WAN IP Deployment 127
Preconfiguring the Cisco ENCS for a Single WAN IP Deployment 128
Single WAN IP Deployment with Gigabit Ethernet Interface 0/0 129
Single WAN IP Deployment with the 4G Interface 130

CHAPTER 13 Resetting to Factory Default 133

CHAPTER 14 NETCONF Event Notifications 135
nfvisEvent 136
vmlcEvent 153

CHAPTER 15 Syslog 171
Syslog Messages 173

CHAPTER 16 SNMP Support on NFVIS 179
Introduction about SNMP 179
SNMP Operations 179
SNMP Get 180
SNMP Notifications 181
SNMP Versions 181
SNMP MIB Support 182
Configuring SNMP Support 186

CHAPTER 17 TACACS and RADIUS Support on NFVIS 193
About RADIUS 193
RADIUS Operation 193
Configuring a TACACS+ Server 194

Configuring RADIUS 195
Default Authentication Order 196

CHAPTER 18 ENCS Switch Configuration 199
ENCS Switch Commands 199
ENCS Switch APIs 199
ENCS Switch Portal Configuration 200
Switch Settings 200
Configuring Spanning Tree 201
Configuring Dot1x 203
Configuring LACP 204
Configuring VLAN 204
Configuring General Settings 205
Configuring Advanced Settings 206
Configuring Spanning Tree per Interface 207

CHAPTER 19 Configuring Secondary IP Address and Source Interface 209

CHAPTER 20 Ports and Port Channels 211
Configuring Port Channels 211
Information About Port Channels 211
Port Channels Bond Mode 212
Port Channels LACP Mode 212
Configuring a Port Channel 212
Adding a Port to a Port Channel 215
Adding a Port Channel to a Bridge 215
Deleting a Port Channel 216
Removing a Port from a Port Channel 216
Removing a Port Channel from a Bridge 216
Configuring LLDP 217
Configuring Administrative Status of a Port 218
Tracking Changes for a Port 218
Speed, Duplex and Autonegotiation 219

CHAPTER 21 MSTP for ENCS 5400 8-Port Switch 223

CHAPTER 22 ENCS 5400 Switch LLDP 225

Link Layer Discovery Protocol (LLDP) Commands 225

Enabling and Disabling LLDP 225

Configuring LLDP Characteristics 226

CHAPTER 23 Secure Overlay and Single IP Configuration 227

Secure Overlay 227

Single Public IP Address and Secure Overlay 237

Single IP Address Without Secure Overlay 242

CHAPTER 24 Dual WAN Support 245

Bridge IP Configurations 245

Restrictions for Bridge IP Configurations 246

Dual WAN Bridge and DHCP Toggle 246

CHAPTER 25 Switch Port Security 247

Switch Port Security 247

PART II NFVIS Functionality Changes for Cisco SD-WAN Cloud OnRamp for Colocation 251

CHAPTER 26 Default System Configurations 253

SRIOV Support 254

OVS-DPDK Support 255

CHAPTER 27 NFVIS Integration with Docker Container Lifecycle 257

Cisco Colo Manager 257

CCM State Transitions from the Host Side 257

CCM Notifications 258

CCM Recovery 259

Support Commands 259

CHAPTER 28 NFVIS Integration with vManage 263

Establishing DTLS Tunnel with vManage 263

NFVIS Notifications 265

Stats for Host and VM 265

System CLI 265

NFVIS Local Portal 265

Core Allocation for Host and CCM 265

CHAPTER 29 Enhancements to VM Image Packaging 267

NFVIS Specific Enhancements 267

Cisco SD-WAN Cloud OnRamp for Colocation Packaging Enhancements 268

VM Packaging Parameters 269

VM Packaging Utility Usage Examples 270

Packaging a VM 270



About this Document

- About this Document, on page xi
- Audience, on page xii
- Related Documentation, on page xii
- Communications, Services, and Additional Information, on page xii

About this Document

This guide provides information about install and configure Cisco Enterprise Network Function Virtualization Infrastructure Software (Cisco Enterprise NFVIS) on a supported Cisco hardware device. The guide also provides details on virtual machine deployments, configuration of software features, and life cycle management using Representation State Transfer (REST) application programming interface (API).

This guide has been divided into chapters starting from installing to maintenance and troubleshooting.

- Quick Start your ENCS Hardware
- Install and Upgrade Cisco NFVIS
- Manage your Host System
- VM Lifecycle Management
- Plug n Play
- System Logs
- ENCS 5400 Switch Management
- Maintain and Monitor Cisco NFVIS
- Configure vBranch HA
- Cisco ENCS Single WAN IP Deployment Scenarios
- Troubleshooting and Debugging

Audience

This guide is intended for network administrators and operators who are familiar with basic Linux installation and configuration requirements.

Related Documentation

- Cisco Network Function Virtualization Infrastructure Software Getting Started Guide
- API Reference for Cisco Enterprise Network Function Virtualization Infrastructure Software
- Cisco Enterprise Network Function Virtualization Infrastructure Software Configuration Guide, Release 4.x
- Cisco Enterprise Network Function Virtualization Infrastructure Software Command Reference
- Release Notes for Cisco NFV SD-Branch features in Cisco vManage Release 20.12.x
- Design and Deployment Guide of Cisco NFVIS SD-Branch using Cisco SD-WAN Manager
- Cisco Catalyst 8200 Series Edge uCPE Data Sheet
- Cisco Cloud Services Platform 5000 Series Data Sheet
- Cisco 5400 Enterprise Network Compute System Hardware Installation Guide
- Cisco 5400 Enterprise Network Compute System Data Sheet
- Configuration Guide for Cisco Network Plug and Play on Cisco APIC-EM, Release 1.5.x
- Cisco SD-WAN Controller Compatibility Matrix and Recommended Computing Resources, Cisco SD-WAN Release 20.12.x

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business results you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco DevNet.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

About this Document

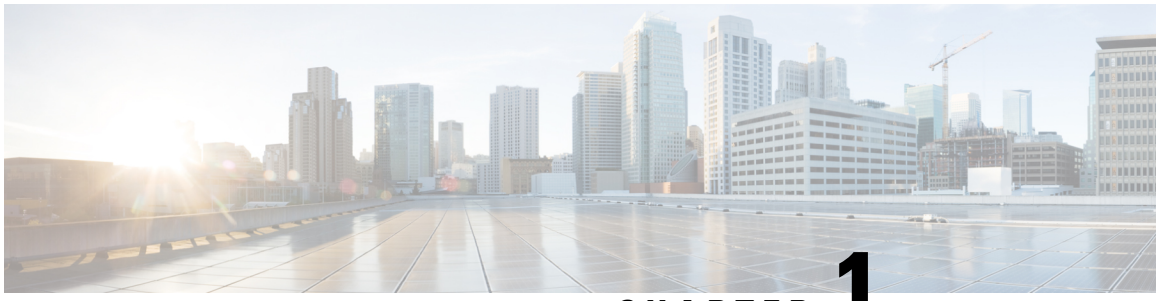


PART I

NFVIS

- About Cisco Enterprise NFVIS, on page 1
- **Installing Cisco Enterprise NFVIS Using the KVM Console** , on page 5
- Setting Up System Configuration, on page 17
- **Cisco Network Plug-n-Play Support** , on page 61
- Overview of VM Life Cycle Management, on page 67
- VM Deployment Scenarios, on page 75
- SPAN Session or Port Mirroring, on page 81
- Configuring Packet Capture, on page 87
- VM Image Packaging, on page 89
- Upgrading Cisco Enterprise NFVIS, on page 109
- Configuring vBranch High Availability, on page 113
- Cisco ENCS Single WAN IP Deployment Scenarios, on page 127
- Resetting to Factory Default, on page 133
- NETCONF Event Notifications, on page 135
- Syslog, on page 171
- SNMP Support on NFVIS, on page 179
- TACACS and RADIUS Support on NFVIS, on page 193
- ENCS Switch Configuration, on page 199
- Configuring Secondary IP Address and Source Interface, on page 209
- Ports and Port Channels, on page 211
- MSTP for ENCS 5400 8-Port Switch, on page 223
- ENCS 5400 Switch LLDP, on page 225
- Secure Overlay and Single IP Configuration, on page 227
- Dual WAN Support, on page 245

• Switch Port Security, on page 247



CHAPTER 1

About Cisco Enterprise NFVIS

Cisco Enterprise Network Function Virtualization Infrastructure Software (Cisco Enterprise NFVIS) is a Linux-based infrastructure software designed to help service providers and enterprises to design, deploy and manage network services. Cisco Enterprise NFVIS helps dynamically deploy virtualized network functions, such as a virtual router, firewall, and WAN accelerator on supported Cisco devices. Such virtualized deployments of VNFs also leads to device consolidation. You no longer need separate devices. Automated provisioning and centralized management also eliminates costly truck rolls.

Cisco Enterprise NFVIS provides a Linux-based virtualization layer to the Cisco Enterprise Network Function Virtualization (ENFV) solution.

Cisco ENFV Solution Overview

The Cisco ENFV solution helps convert your critical network functions into a software which can deploy network services across dispersed locations in minutes. It provides a fully integrated platform that can run on top of a diverse network of both virtual and physical devices with the following primary components:

- Cisco Enterprise NFVIS
- VNFs
- Unified Computing System (UCS) and Enterprise Network Compute System (ENCS) hardware platforms
- Digital Network Architecture Center (DNAC)

- Benefits of Cisco Enterprise NFVIS, on page 1
- Supported Hardware Platforms, on page 2
- Supported VMs, on page 3
- Key Tasks You can Perform Using Cisco Enterprise NFVIS, on page 4

Benefits of Cisco Enterprise NFVIS

- Consolidates multiple physical network appliances into a single server running multiple virtual network functions.
- Deploys services quickly and in a timely manner.
- Cloud based VM life cycle management and provisioning.
- Life cycle management to deploy and chain VMs dynamically on the platform.

- Programmable APIs.

Supported Hardware Platforms

Depending on your requirement, you can install Cisco Enterprise NFVIS on the following Cisco hardware platforms:

- Cisco 5100 Series Enterprise Network Compute System (Cisco ENCS)
- Cisco 5400 Series Enterprise Network Compute System (Cisco ENCS)
- Cisco Catalyst 8200 Series Edge Universal CPE
- Cisco Catalyst 8300 Series Edge Universal CPE
- Cisco UCS C220 M4 Rack Server
- Cisco UCS C220 M5Rack Server
- Cisco UCS C M6 Rack Servers (UCSC-C220-M6S, UCSC-C240-M6SX, and UCSC-C240-M6S)
- Cisco Cloud Services Platform 2100 (CSP 2100)
- Cisco Cloud Services Platform 5228 (CSP-5228), 5436 (CSP-5436) and 5444 (CSP-5444 Beta)
- Cisco ISR4331 with UCS-E140S-M2/K9
- Cisco ISR4351 with UCS-E160D-M2/K9
- Cisco ISR4451-X with UCS-E180D-M2/K9
- Cisco UCS-E160S-M3/K9 Server
- Cisco UCS-E180D-M3/K9
- Cisco UCS-E1120D-M3/K9

Cisco ENCS

The Cisco 5100 and 5400 Series Enterprise Network Compute System combines routing, switching, storage, processing, and a host of other computing and networking activities into a compact one Rack Unit (RU) box. This high-performance unit achieves this goal by providing the infrastructure to deploy virtualized network functions and acting as a server that addresses processing, workload, and storage challenges.

Cisco Catalyst 8200 Series Edge Universal CPE

The Cisco Catalyst 8200 Edge uCPE is the next generation of Cisco Enterprise Network Compute System 5100 Series that combines routing, switching and application hosting into a compact one rack unit device for the small and Medium Virtualized Branch. These platforms are designed to allow customers to run virtualized network functions and other applications as virtual machines on the same hardware platform powered by Cisco NFVIS hypervisor software. These devices are 8 Core x86 CPUs with HW Acceleration for IPSec crypto traffic with higher number of WAN ports. They have a NIM slot and a PIM slot to choose different WAN, LAN and LTE/5G modules for the Branch.

Cisco Catalyst 8300 Series Edge Universal CPE

The Cisco Catalyst 8300 Series Edge Universal Customer Premises Equipment (uCPE) is a purpose-built x86 platform that is designed for branch virtualization. It enables device consolidation across network and security functions, improves operational flexibility and service agility, simplifies network operations, and results in reduced deployment times and fewer truck rolls for delivery of add-on services.

Cisco UCS C220 M4/M5 Rack Server

The Cisco UCS C220 M4 Rack Server is a high-density, general-purpose enterprise infrastructure and application server that delivers world class performance for a wide range of enterprise workloads, including virtualization, collaboration, and bare-metal applications.

Cisco UCS C M6 Rack Server

The Cisco UCS C220 M6 Rack Server is the most versatile general-purpose infrastructure and application server in the industry. This high-density, 1RU, 2-socket rack server delivers industry-leading performance and efficiency for a wide range of workloads, including virtualization, collaboration, and bare-metal applications.

Cisco CSP 2100-X1, 5228, 5436 and 5444 (Beta)

Cisco Cloud Services Platform is a software and hardware platform for data center network functions virtualization. This open kernel virtual machine (KVM) platform is designed to host networking virtual services. Cisco Cloud Services Platform devices enables network, security, and load balancer teams to quickly deploy any Cisco or third-party network virtual service.

Note CSP 5000 series devices support ixgbe drivers.

Caution If CSP platforms are running NFVIS, Return Material Authorization (RMA) is not supported.

Cisco UCS E-Series Server Modules

The Cisco UCS E-Series Servers (E-Series Servers) are the next generation of Cisco UCS Express servers. E-Series Servers are a family of size, weight, and power efficient blade servers that are housed within the Generation 2 Cisco Integrated Services Routers (ISR G2), Cisco 4400, and Cisco 4300 Series Integrated Services Routers. These servers provide a general-purpose compute platform for branch office applications deployed either as bare metal on operating systems, such as Microsoft Windows or Linux; or as virtual machines on hypervisors.

Supported VMs

Currently, Cisco Enterprise NFVIS supports the following Cisco VMs and third-party VMs:

- Cisco Catalyst 8000V Edge Software
- Cisco Integrated Services Virtual (ISRv)
- Cisco Adaptive Security Virtual Appliance (ASAv)

- Cisco Virtual Wide Area Application Services (vWAAS)
- Linux Server VM
- Windows Server 2012 VM
- Cisco Firepower Next-Generation Firewall Virtual (NGFWv)
- Cisco vEdge
- Cisco XE SD-WAN
- Cisco Catalyst 9800 Series Wireless Controller
- ThousandEyes
- Fortinet
- Palo Alto
- CTERA
- InfoVista

Key Tasks You can Perform Using Cisco Enterprise NFVIS

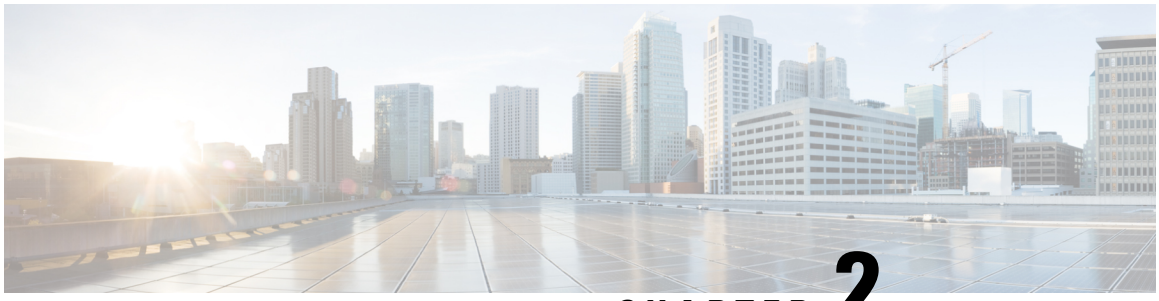
- Perform VM image registration and deployment
- Create new networks and bridges, and assign ports to bridges
- Perform service chaining of VMs
- Perform VM operations
- Verify system information including CPU, port, memory, and disk statistics
- SR-IOV support on all interfaces of all platforms, with the exception of UCS-E backplane interface

The APIs for performing these tasks are explained in the API Reference for Cisco Enterprise NFVIS.



Note NFVIS can be configured through Netconf interface, REST APIs and command-line interface as all the configurations are exposed through YANG models.

From a Cisco Enterprise NFVIS command-line interface, you can connect to another server and VMs remotely using the SSH client.



CHAPTER 2

Installing Cisco Enterprise NFVIS Using the KVM Console

- Installation Prerequisites , on page 5
- Image Signing and Verification, on page 6
- Entering BIOS Setup, on page 7
- Installing Cisco Enterprise NFVIS on the Cisco UCS C220 M4 and M5 Rack Servers or Cisco CSP Devices, on page 7
- Installing Cisco Enterprise NFVIS on Cisco UCS E-Series Servers, on page 8
- Installing Cisco Enterprise NFVIS on a Cisco ENCS 5100 and 5400, on page 12

Installation Prerequisites

Ensure that the following prerequisties are met:

- The IP address is configured for Cisco Integrated Management Controller (CIMC) as well as a login account with administrative privileges.
- The Cisco Enterprise NFVIS ISO image is available.
- The IP address of the system (required for remote access) is available.
- Hyper-threading is enabled in BIOS. By default, hyper-threading is enabled in BIOS on the UCS-C, UCS-E and ENCS platforms.

Note The installation steps are slightly different for Cisco UCS and Cisco ENCS platforms. See the following sections for details:

Installing Cisco Enterprise NFVIS on the Cisco UCS C220 M4 and M5 Rack Servers or Cisco CSP Devices, on page 7

Installing Cisco Enterprise NFVIS on Cisco UCS E-Series Servers, on page 8

Installing Cisco Enterprise NFVIS on a Cisco ENCS 5100 and 5400, on page 12

Assumptions

- The user is familiar with the supported hardware device, CIMC, Cisco Network Plug and Play, and Cisco Application Policy Infrastructure Controller Enterprise Module (APIC-EM).
- The initial setup of the hardware device is complete, and the device is ready for loading Cisco Enterprise NFVIS.
- The user is familiar with general Linux installation.

For more details on the supported hardware devices, see respective documentation available on Cisco.com.

Image Signing and Verification

Cisco Enterprise NFVIS supports RPM signing and signature verification for all RPM packages in the ISO and upgrade images. You can also verify the integrity of the Cisco Enterprise NFVIS 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 Cisco Enterprise NFVIS. The private key, used for signing the RPM packages, is created and securely maintained by Cisco.

RPM Signature Verification

Cisco Enterprise NFVIS verifies all RPM packages during installation or upgrade. The following table describes the Cisco Enterprise NFVIS behavior when the signature verification fails during installation or upgrade.

Scenario	Description
Cisco Enterprise NFVIS 3.7.1 installation	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 Using sha256sum

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 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/sha256sum ImageFile
4db533d96d8705db8af904ab754349151adea504b81337155cc591c6203e3295 ImageFile
```

Entering BIOS Setup

Note This section applies only to ENCS 5400 and UCS-E series routers.

When you enter the BIOS setup for the first time, ensure that you secure the BIOS by setting up an admin-level and a user-level password. You have to set up the admin password when you access the BIOS menu for the first time. The user password (which only gives access to a small subset of BIOS options) must be set inside the BIOS setup menu.

To set up the admin password, press F2 when the system boots up. You will be prompted to set the password.

To set up the user password, after you log in, go to the 'Security' tab and set the password.

Installing Cisco Enterprise NFVIS on the Cisco UCS C220 M4 and M5 Rack Servers or Cisco CSP Devices

This section provides information about a series of tasks you need to perform to install Cisco Enterprise NFVIS on a Cisco UCS C220 M4 and M5 Rack Server or Cisco CSP 2100-X1, CSP 5228 and CSP 5436.

Logging Into the CIMC GUI

Before you begin

- Ensure that you have configured the IP address to access CIMC as well as a login account with administrative privileges.
- The recommended CIMC version is 3.0(3c) or newer.
- Ensure that a Virtual Drive is created with RAID and Boot Drive is set to true.

For details on how to configure an IP address for CIMC, see the Set up CIMC for UCS C-Series Server guide on cisco.com.

Procedure

Step 1 Log into CIMC.

Step 2 Launch KVM Console:

From theCIMC Homepage, click Launch KVM on the upper right corner.You can choose from Java based KVM or HTML based KVM. It is recommnded to use HTML based KVM. Ensure that the pop-up blocker is disabled. KVM Console will open in a separate window.

Step 3 Map virtual media:

a) From KVM Console, click on **Virtual Media** and then click on **Activate Virtual Devices**.
b) Click on **Virtual Media** again and then click on **Map CD/DVD**, browse, and select the Cisco Enterprise NFVIS ISO image, click **Open** and Map Drive to mount the image.
c) Click on **Virtual Media** again to ensure the NFVIS ISO image is now mapped to CD/DVD.

Step 4 Configure Boot Order:

a) From the **CIMC Compute**, select **BIOS**.
b) Select **Configure Boot Order** and click on **Configure Boot Order**.

   The **Configure Boot Order** dialog box appears.

c) Select **Advanced**.
d) From **Add Boot Device**, select **Add Virtual Media**, and the **Add Virtual Media** dialog box appears.
e) Enter a name, select **KVM Mapped DVD**, set state to **Enabled** and order to 1, click on **Save Changes**.
f) From **Add Boot Device**, select **Add Local HDD**, the **Add Virtual Media** dialog box appears.
g) Enter a name, set Sstate to Enabled and order to 2, click on **Save Changes**.
h) Click **Close**.

Step 5 Power cycle server to start the installation:

From CIMC Home page, click on **Host Power** on the upper right corner. Reboot the server by selecting the **Power Off** option. After the server is down, select the **Power On** option.

When the server reboots, the KVM console automatically installs Cisco Enterprise NFVIS from the virtual CD/DVD drive. The entire installation might take 30 minutes to one hour to complete.

Step 6 Initial login: After the installation is complete, the system automatically reboots from the hard drive. Log into the system when the command prompt **nfvis login** is displayed after the reboot.

Use **admin** as the login name and **Admin123#** as the default password.

Note
The system prompts you to change the default password at the first login attempt. You must set a strong password as per the on-screen instructions to proceed with the application. You cannot run API commands or proceed with any tasks unless you change the default password at the first login. API will return 401 unauthorized error if the default password is not reset.

Step 7 Verification: You can verify the installation using the System API, CLI, or by viewing the system information from the Cisco Enterprise NFV portal.

Installing Cisco Enterprise NFVIS on Cisco UCS E-Series Servers

Before you begin

- Configure the Gigabit Ethernet interface on the Cisco ISR router.
- Configure the UCS E interface on the Cisco ISR router.
- Ensure that you have the IP address configured for CIMC access as well as a login account with administrative privileges.

- The recommneded CIMC version is 3.2(7) or newer.
- Ensure that a Virtual Drive is created with RAID and Boot Drive is set to true.
- Ensure that the Cisco UCS E server has one of the following supported firmware versions or above:
  - BIOS UCSED.2.5.0.3 or later for UCS-E160D-M2/K9 and UCS-E180D-M2/K9
  - BIOS UCSES.1.5.0.5 or later for UCS-E140S-M2/K9

For more details on how to perform the basic configuration on the Cisco ISR routers, see the following guides:

- Sample Configuration on the Cisco ISR Router to Bring Up a Cisco UCS E Server, on page 10
- Getting Started Guide for Cisco UCS E-Series Servers, Release 1.0(2) Installed in the Cisco ISR 4451-X

For details on how to configure an IP address for CIMC, see the Getting Started Guide for Cisco UCS E-Series Servers, Release 1.0 on cisco.com.

Procedure

Step 1 Log into CIMC.

Step 2 Launch KVM Console:

From theCIMC Homepage, click Launch KVM on the upper right corner.You can choose from Java based KVM or HTML based KVM. It is recommnded to use HTML based KVM. Ensure that the pop-up blocker is disabled. KVM Console will open in a separate window.

Step 3 Map virtual media:

a) From KVM Console, click on **Virtual Media** and then click on **Activate Virtual Devices**.
b) Click on **Virtual Media** again and then click on **Map CD/DVD**, browse, and select the Cisco Enterprise NFVIS ISO image, click **Open** and Map Drive to mount the image.
c) Click on **Virtual Media** again to ensure the NFVIS ISO image is now mapped to CD/DVD.

Step 4 Configure Boot Order:

a) From the **CIMC Compute**, select **BIOS**.
b) Select **Configure Boot Order** and click on **Configure Boot Order**.

   The **Configure Boot Order** dialog box appears.

c) From the **CD/DVD** area, select **Cisco vKVM-Mapped vDVD**, and then click **Add**.
d) From **HDD**, select **RAID Adapter**, and then click **Add**.
e) Set the boot order sequence using the **Up** and **Down** options. The Cisco vKVM-Mapped vDVD boot order option must be the first choice. Click **Save Changes** to complete the boot order setup.

Step 5 Power cycle server to start the installation:

From CIMC Home page, click on **Host Power** on the upper right corner. Reboot the server by selecting the **Power Off** option. After the server is down, select the **Power On** option.

When the server reboots, the KVM console automatically installs Cisco Enterprise NFVIS from the virtual CD/DVD drive. The entire installation might take 30 minutes to one hour to complete.

Step 6 Initial login: After the installation is complete, the system automatically reboots from the hard drive. Log into the system when the command prompt **nfvis login** is displayed after the reboot.

Use **admin** as the login name and **Admin123#** as the default password.

Note
The system prompts you to change the default password at the first login attempt. You must set a strong password as per the on-screen instructions to proceed with the application. You cannot run API commands or proceed with any tasks unless you change the default password at the first login. API will return 401 unauthorized error if the default password is not reset.

Step 7 Verification: You can verify the installation using the System API, CLI, or by viewing the system information from the Cisco Enterprise NFV portal.

What to do next

You can verify the default configuration, and set up initial IP configuration to launch the Cisco Enterprise NFV portal. For details, see Performing Initial System Configuration.

Sample Configuration on the Cisco ISR Router to Bring Up a Cisco UCS E Server

The following sample configuration shows the basic configuration performed on the Cisco ISR 4451 router with DHCP enabled.

```
Last configuration change at 02:36:37 UTC Thu Feb 18 2016
!
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no platform punt-keepalive disable-kernel-core
!
hostname NFVIS-ISR4451
!
boot-start-marker
boot system bootflash:isr4300-universalk9.03.16.01a.S.155-3.S1a-ext.SPA.bin
boot-end-marker
!
!
vrf definition Mgmt-intf
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
!
!
no aaa new-model
!
!
!
ip domain name cisco.com
!
!
!
```

```
subscriber templating
!
multilink bundle-name authenticated
!
!
!
license udi pid ISR4331/K9 sn FDO192207MN
!
!
ucse subslot 1/0
 imc access-port shared-lom console
 imc ip address 172.19.183.172 255.255.255.0 default-gateway 172.19.183.1
!
spanning-tree extend system-id
!
!
redundancy
 mode none
!
!
!
vlan internal allocation policy ascending
!
!
!
interface GigabitEthernet0/0/0
 ip address 172.19.183.171 255.255.255.0
 media-type rj45
 negotiation auto
!
interface GigabitEthernet0/0/1
 no ip address
 shutdown
 negotiation auto
!
interface GigabitEthernet0/0/2
 no ip address
 shutdown
 negotiation auto
!
interface ucse1/0/0
 ip unnumbered GigabitEthernet0/0/0
 negotiation auto
 switchport mode trunk
 no mop enabled
 no mop sysid
!
interface ucse1/0/1
 no ip address
 no negotiation auto
 switchport mode trunk
 no mop enabled
 no mop sysid
!
interface GigabitEthernet0
 vrf forwarding Mgmt-intf
 no ip address
 shutdown
 negotiation auto
!
interface Vlan1
 no ip address
 shutdown
!
```

```
ip default-gateway 172.19.183.1
ip forward-protocol nd
no ip http server
no ip http secure-server
ip tftp source-interface GigabitEthernet0
ip route 0.0.0.0 0.0.0.0 172.19.183.1
ip route 172.19.183.172 255.255.255.255 ucse1/0/0
ip ssh version 2
!
!
!

control-plane
!
!
line con 0
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 password lab
 login local
 transport input all
 transport output all
!
!
end
```

Installing Cisco Enterprise NFVIS on a Cisco ENCS 5100 and 5400

Note Software or hardware RAID controller setup is not supported with Cisco ENCS in Cisco Enterprise NFVIS Release 3.5.1.

Before you begin

- Make sure that you have configured the IP address to access CIMC.
- The recommended CIMC version is 3.2(7) or newer.

For details on how to configure an IP address for CIMC, see the Set up CIMC for UCS C-Series Server and Getting Started Guide for Cisco UCS E-Series Servers and the Cisco UCS E-Series Network Compute Engine on cisco.com.

Procedure

Step 1 Log into CIMC.

Step 2 Launch KVM Console:

From theCIMC Homepage, click Launch KVM on the upper right corner.You can choose from Java based KVM or HTML based KVM. It is recommnded to use HTML based KVM. Ensure that the pop-up blocker is disabled. KVM Console will open in a separate window.

Step 3 Map virtual media:

a) From KVM Console, click on **Virtual Media** and then click on **Activate Virtual Devices**.
b) Click on **Virtual Media** again and then click on **Map CD/DVD**, browse, and select the Cisco Enterprise NFVIS ISO image, click **Open** and Map Drive to mount the image.
c) Click on **Virtual Media** again to ensure the NFVIS ISO image is now mapped to CD/DVD.

Step 4 Configure Boot Order:

a) From the **CIMC Compute**, select **BIOS**.
b) Select **Configure Boot Order** and click on **Configure Boot Order**.

   The **Configure Boot Order** dialog box appears.

c) From the **CD/DVD** area, select **Cisco vKVM-Mapped vDVD**, and then click **Add**.
d) From **HDD**, select **RAID Adapter**, and then click **Add**.
e) Set the boot order sequence using the **Up** and **Down** options. The Cisco vKVM-Mapped vDVD boot order option must be the first choice. Click **Save Changes** to complete the boot order setup.

Step 5 Power cycle server to start the installation:

From CIMC Home page, click on **Host Power** on the upper right corner. Reboot the server by selecting the **Power Off** option. After the server is down, select the **Power On** option.

When the server reboots, the KVM console automatically installs Cisco Enterprise NFVIS from the virtual CD/DVD drive. The entire installation might take 30 minutes to one hour to complete.

Step 6 Firmware Auto-Upgrade.

Starting from NFVIS 3.8.x release, firmware auto-upgrade is supported. After the NFVIS installation is complete, BIOS and/or CIMC is upgraded to the corresponding versions automatically. CIMC and NFVIS is rebooted multiple times. The firmware upgrade might take 30 minutes to one hour to complete. Do not use the system during the firmware upgrade.

Step 7 Initial login: After the installation is complete, the system automatically reboots from the hard drive. Log into the system when the command prompt **nfvis login** is displayed after the reboot.

Use **admin** as the login name and **Admin123#** as the default password.

Note
The system prompts you to change the default password at the first login attempt. You must set a strong password as per the on-screen instructions to proceed with the application. You cannot run API commands or proceed with any tasks unless you change the default password at the first login. API will return 401 unauthorized error if the default password is not reset.

Step 8 Verification: You can verify the installation using the System API, CLI, or by viewing the system information from the Cisco Enterprise NFV portal.

Install Cisco Enterprise NFVIS on Cisco ENCS 5104

Procedure

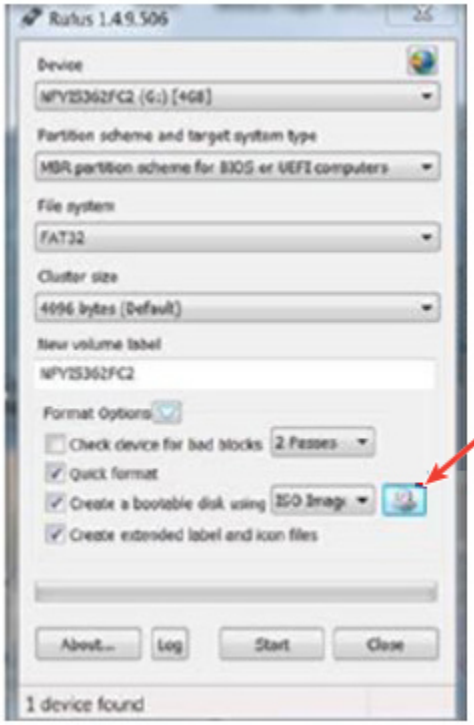
Step 1 Create bootable usb with NFVIS image.

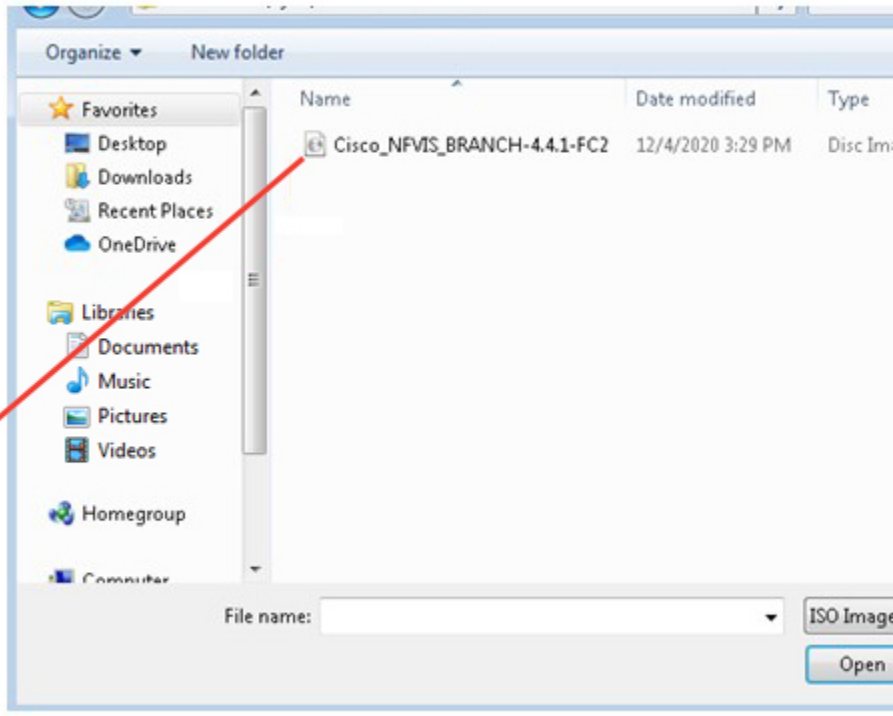
In this example, we used rufus utility in Windows environment. Rufus utility can be downloaded https://rufus.akeo.ie/. For this example, following parameters were used to burn bootable NFVIS USB device:

- Device: USB stick
- Partition scheme: MBR
- Filesystem: FAT32
- Cluster size: use default
- Volume label: use default
- Quick format: checked
- Create bootable: select "ISO Image" and click next icon then choose NFVIS image.
- Create extended label: checked

Press **Start** and wait for completion.

Eject USB thumb drive





Step 2 Insert USB device in one of USB slot in ENCS5104.

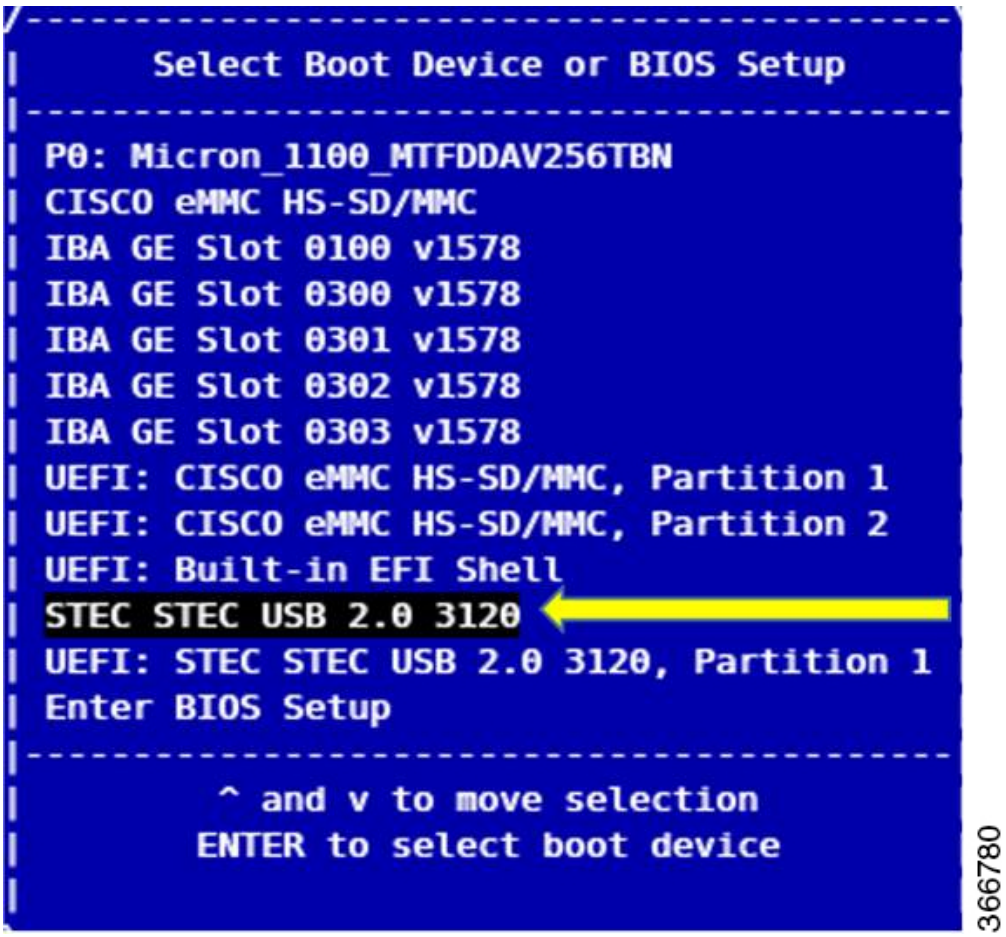
Step 3 Power on system.

Step 4 During system boot up, press F6 key.

```
Press <DEL> or <F2> to enter setup, <F6> Boot Menu, <F12> Network Boot in 5 seconds or press any key
 to continue.
```

Step 5 Once you press F6, you will see the following screenshot to select which device you want to boot from. Select your USB device.

In the following screenshot example, there is STEC USB being used. That display will vary depending on your usb device vendor. Use the arrow key to select that device.



Step 6 Wait until installation is completed. System will be rebooted once installation is done.

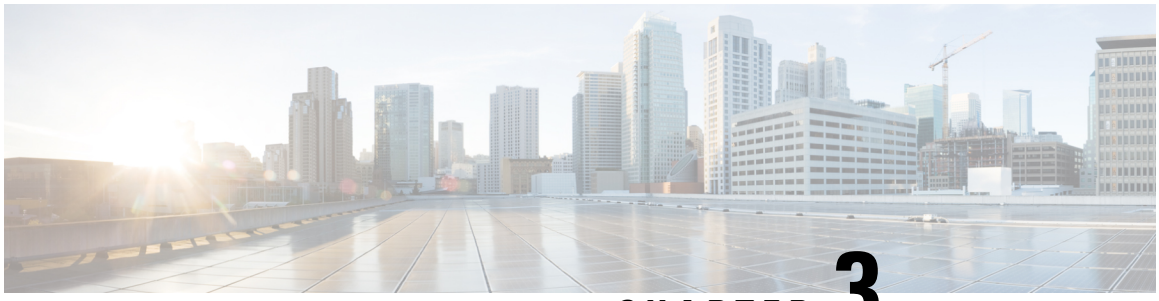
Step 7 Log into the system with username **admin** and **Admin123#** as a default password

Step 8 You will be prompted and asked to change password at the first login. You must set a strong password per the on-screen instruction to proceed.

Step 9 You can verify the installation status using the System API or command line interface per the NFVIS user guide.

What to do next

You can verify the default configuration, and set up initial IP configuration to launch the Cisco Enterprise NFV portal. For details, see Performing Initial System Configuration.



CHAPTER 3

Setting Up System Configuration

- Default System Configuration on the Cisco ENCS, on page 17
- Default System Configuration on the Cisco UCS C220 M4 Server and Cisco CSP 2100, on page 19
- Default System Configuration on the Cisco UCS E-Series Servers , on page 20
- Accessing NFVIS, on page 20
- User Roles and Authentication, on page 27
- Configuring the IP Receive ACL, on page 31
- Configuring Your Banner and Message of the Day, on page 32
- Setting the System Time Manually or With NTP, on page 33
- Enable or Disable Access to NFVIS Portal, on page 34
- Configuring System Logs, on page 35
- Network File System Support, on page 36
- Secure Boot of host, on page 37
- CIMC Access Control, on page 38
- DPDK Support on NFVIS, on page 42
- Import and Export NFVIS VM, on page 45
- Backup and Restore NFVIS and VM Configurations, on page 48
- Dynamic SR-IOV, on page 57

Default System Configuration on the Cisco ENCS

The diagram below illustrates the default network configuration of Cisco Enterprise NFVIS with the Cisco ENCS.

Figure 1: Default Network Configuration of Cisco Enterprise NFVIS with the Cisco ENCS 5400

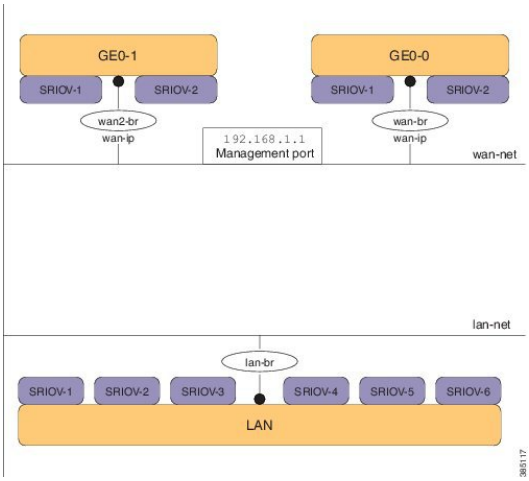
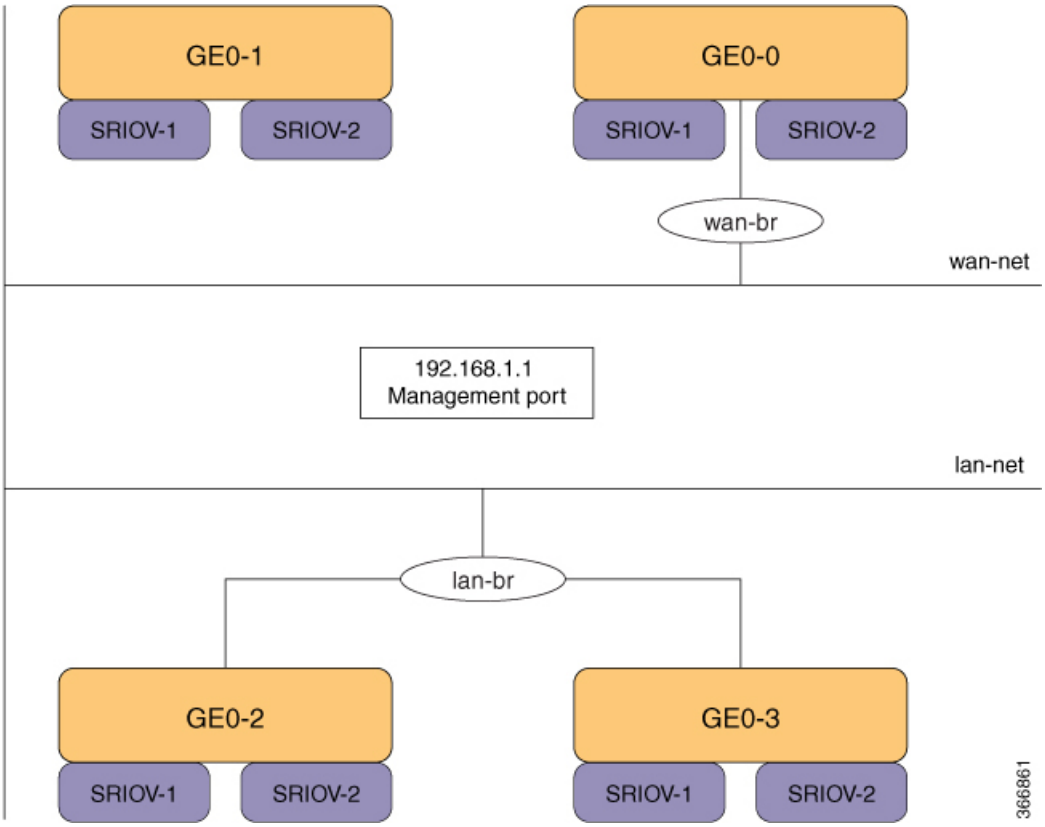


Figure 2: Default Network Configuration of Cisco Enterprise NFVIS with the Cisco ENCS 5100



- LAN ports—Eight physical Gigabit Ethernet ports for inbound and outbound traffic.
- WAN port—You can use one of the dual media Ethernet ports (wan-br and wan2-br) for DHCP connection.
- Bridges—They form a Layer 2 domain between virtual network interface controllers (vNICs) of VMs. A vNIC is used by a virtual machine to provide virtual network interfaces by defining a range of MAC

addresses. The default management IP address (192.168.1.1) for the NFVIS host is configured on the management port. Multiple VMs can use the same LAN port for local connectivity.

- Network—It is a segment Layer 2 bridge domain where only the specific VLAN traffic is allowed.
- Reserved VLANs in the LAN network on the ENCS 5400 platform—The VLAN range 2350-2449 is reserved for internal use and should not be used on the external switch ports and for virtual machines in the LAN ports". Note that this limitation doesn't apply to the WAN ports.
- Internal 192.168.10.00/24 and 192.168.50.0/24 networks—The IP subnet 192.168.10.0/24 and 192.168.50.0/24 are used for the ENCS-5400 internal networks. A user should not use this IP subnet on the NFVIS management network. In the future NFVIS releases, this internal subnet will be isolated so that users can use this for NFVIS management.

Note The following networks and bridges are automatically configured. You can configure more as required.

- A LAN network (lan-net) and a LAN bridge (lan-br)
- A WAN network (wan-net) and a WAN bridge (wan-br)

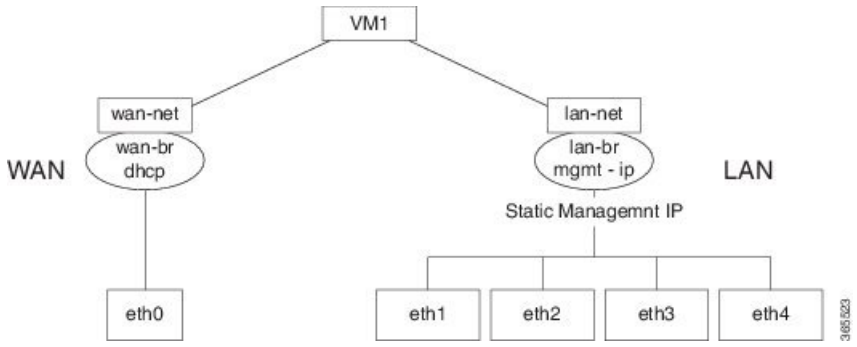
wan2-net and wan2-br are the default configurations for ENCS 5400 and ENCS 5100.

The default networks and bridges cannot be deleted.

Default System Configuration on the Cisco UCS C220 M4 Server and Cisco CSP 2100

Configuring the networks in Cisco Enterprise NFVIS allows inbound and outbound traffic and VMs to be service chained. The following diagram illustrates the default network configuration:

Figure 3: Default Network Configuration with Cisco UCS C220 M4 and Cisco CSP 2100



The following networks and bridges are created by default, and cannot be deleted. You can configure more as required.

- A LAN network (lan-net) and a LAN bridge (lan-br)—The default static management IP address (192.168.1.1) for the NFVIS host is configured on the LAN bridge. One of the ports for inbound and

outbound traffic are associated with the LAN bridge. Any LAN port can be used to access the default static IP address. By default, the hostname is set to "nfvis".

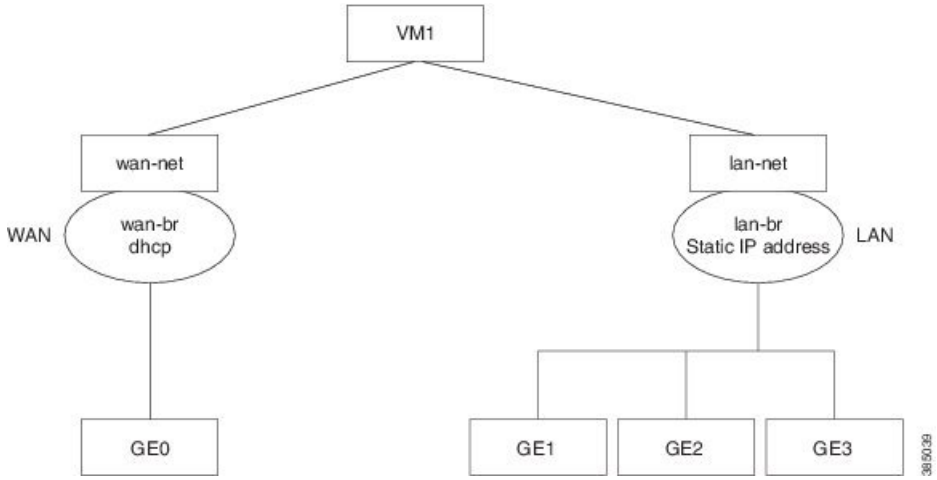
- A WAN network (wan-net) and a WAN bridge (wan-br)—This is created with the "eth0" port, and is configured to enable the DHCP connection.

By default, the first port on the device is associated with the WAN bridge. One of the other ports on the device are associated with the LAN bridge.

For more details about the initial setup, see the Installing the Server chapter in the *Cisco UCS C220 M4 Server Installation and Service Guide* or *Cisco Cloud Services Platform 2100 Hardware Installation Guide*.

Default System Configuration on the Cisco UCS E-Series Servers

Figure 4: Default Network Configuration with a Cisco UCS E-Series Server



The following networks and bridges are created by default, and cannot be deleted. You can configure more as required.

- A LAN network (lan-net) and a LAN bridge (lan-br)—The default static management IP address (192.168.1.1) for the NFVIS host is configured on the LAN bridge. All other ports for inbound and outbound traffic are associated with the LAN bridge. By default, the hostname is set to "nfvis".
- A WAN network (wan-net) and a WAN bridge (wan-br)— The physical WAN ports are on the Cisco ISR module. They are not externally available on the Cisco UCS E server. The WAN traffic comes from the ISR WAN ports, and goes through the backplane to the Cisco UCS-E server. The backplane has one internal WAN interface (GE0) to establish connection with the Cisco UCS-E server. By default, the "GE0" interface is enabled for the DHCP connection.

For more details on the initial setup, see the Getting Started Guide for Cisco UCS E-Series Servers and the Cisco UCS E-Series Network Compute Engine.

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 returns 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 61).
- Using console (After the initial login using the default password, you are prompted to change the default password).

```
NFVIS Version: 3.10.0-9

Copyright (c) 2015-2018 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.

nfvis login: console (automatic login)

login:
login:
login:
login:
login: admin

Cisco Network Function Virtualization Infrastructure Software (NFVIS)

NFVIS Version: 3.10.0-9

Copyright (c) 2015-2018 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@localhost's password:
```

```
admin connected from ::1 using ssh on nfvis
nfvis# show version


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.

Connect 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 a static IP address of 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. However, to be able to use a static IP address to remotely connect to a server, the default gateway needs to be configured first.

You can connect to the system in the following 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 the Managed (M) flag, then the network stays in DHCP mode, even if there is no DHCP server in the network. If the RA shows the 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.

Perform 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 the **no bridges bridge wan2-br dhcp** command.

If you want to disable DHCP and use static configuration, you need to perform the initial configuration 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.

Secure overlay is not supported when WAN interface is configured with IPv6.

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. You 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
```

Verify Initial Configuration

Use the **show system settings-native** command 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.

Here is an 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
```

Here is an 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
```

Here is an 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 creates 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 bridges and LAN bridges 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

User Roles and Authentication

Role based access enables the administrator to manage different levles 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 1: 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.

To create a user:

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

To delete a user:

```
configure terminal
rbac authentication users delete-user name test1
commit
```

Note To change the password, use the **rbac authentication users user change-password** command in global configuartion mode. To change the user role, use the **rbac authentication users user change-role** command in global configuration mode.

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 /oper/change-role • /api/config/rbac/authentication/users/user?deep	• rbac authentication users • rbac authentication users user change-password • rbac authentication users user change-role

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/	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

Certification

Generate Sign-Request

```
nfvis(config)# system certificate signing-request ?

Possible completions:

  common-name             country-code

  locality                organization

  organization-unit-name  state
```

The .csr file will be saved in /data/intdatastore/download/nfvis.csr

Use the scp command to download the file.

Install CA Sign Certificate

After CA sign in, the user needs to use the scp command to upload the file into nfvis.

```
nfvis(config)# system certificate install-cert path file:///<full path of the file>
```

The path needs to start with "file://"

Switch Certificate

```
nfvis(config)# system certificate use-cert cert-type ca-signed
```

nginx process restarts after the switch.

The users cannot access the log files. The log files are added to all the user actions and the user can download and view some of the logs from portal. A notification is generated when the log files reach 75% capacity.

Secure Copy Command

The secure copy (**scp**) command allows only the admin user to secure copy a file from the Cisco NFVIS to an external system or from an external system to Cisco NFVIS. The **scp** 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.

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
```

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 users with administrator priviledges can use the SCP command on port 22222 to upload or download only from restricted folders like /data/intdatastore/. For more information, see Host System Operations.



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 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 is not 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

Enable or Disable Access to NFVIS Portal

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

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

Network File System Support

Network File System (NFS) is an application where you can view, store, and update the files on a remote device. NFS allows you 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.

Mount and Unmount NFS

The following example shows how to mount NFS:

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

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

Image Registration on NFS

Images in tar.gz, ISO and qcow2 formats, 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 the **placement type zone_host host nfs_storage** command.

Secure Boot of host



Note This feature is available only for NFVIS 3.9.1 release fresh install and supported only on ENCS 5400. Upgrade BIOS to version 2.6 for this feature.

The secure boot feature prevents malicious software applications and unauthorized operating systems from loading into the system during the system start up process. If secure boot feature is enabled, only the authorized software applications boots up from the device. Each device has keys that allow software with the correct signature to boot up on the device.

This feature ensures that the software applications that boot up on the device are certified by Cisco. The NFVIS 3.9.1 image is signed with Cisco key. If secure boot is enabled the signature is verified during the device boot up. If the verification fails, the image does not boot up.

Secure boot is disabled by default and to enable it you must change firmware configurations from CIMC. Secure boot needs to boot from a separate UEFI partition.

To enable secure boot:

1. Access CIMC and use **show bios detail** command to view the BIOS version.

```
ENCS# scope bios
ENCS/bios # show detail
BIOS:
    BIOS Version: " ENCS54_2.6 (Build Date: 07/12/2018)"
    Boot Order: EFI
    FW Update/Recovery Status: Done, OK
    Active BIOS on next reboot: main
    UEFI Secure Boot: disabled
ENCS/bios #
```

2. Enable secure boot of host.

```
ENCS/bios # set secure-boot enable
Setting Value : enable
Commit Pending.
ENCS/bios *# commit
ENCS/bios # show detail
BIOS:
    BIOS Version: "ENCS54_2.6 (Build Date: 07/12/2018)"
    Boot Order: EFI
    FW Update/Recovery Status: None, OK
    Active BIOS on next reboot: main
    UEFI Secure Boot: enabled
ENCS/bios #
```

Legacy boot, UEFI boot and UEFI secure boot are the three boot modes. Secure boot can only be used on a disk that has UEFI partition.

You can configure boot order from CIMC command or portal or from BIOS setup menu. You can only configure legacy boot order with CIMC . By default, **BootOrderRules** are set to **Strict**, so the boot order follows the CIMC configuration. Since CIMC cannot be used to configure UEFI boot order, to enable secure boot change the **BootOrderRules** setting to **Loose**.

If **BootOrderRules** is set to **Loose**, the boot order follows the BIOS setup menu. When an operating system is installed in secure boot mode, the new UEFI boot option for the OS automatically appears at the top of the BIOS menu boot order list, to boot the installed operating system.

To set **BootOrderRules** to **Loose**:

```
ENCS/bios # scope advanced
ENCS/bios/advanced # set BootOrderRules Loose
ENCS/bios/advanced *# commit
Changes to BIOS set-up parameters will require a reboot.
Do you want to reboot the system?[y|N]y
```

CIMC Access Control

NFVIS administrators have authoritative control control over ENCS 5400 devices. 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

For releases 3.8.1 and later, if the BIOS or CIMC versions on Cisco ENCS 5400 routers are lower than the image version in the NFVIS ISO or upgrade package, the BIOS and CIMC versions on the routers are automatically upgraded to the version of the bundled image during NFVIS upgrade or installation. The CPU microcode is also upgraded as part of this upgrade or installation. Note that the upgrade process takes longer than in previous releases and the process cannot be stopped midway.

For Cisco ENCS 5100 routers, BIOS is automatically upgraded to the new version, but the server needs to be rebooted manually for the upgrade to show.

BIOS and CIMC Password

Table 2: Feature History

Feature Name	Release Information	Description
BIOS and CIMC password	NFVIS 4.2.1	New password restrictions and security measures are added for CIMC and BIOS.

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.



Note New password restrictions added for CIMC and BIOS in NFVIS 4.2.1 release.

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

- Must contain at least one upper case and one lower case letter.
- Must contain at least one number and one special character from #, @ or _.
- Length should be between 8 and 20 characters.
- Should not contain the following string (case sensitive): admin

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

- The first letter cannot be #.
- Must contain at least one upper case and one lower case letter.
- Must contain at least one number and one special character from #, @ or _.
- Length should be between 8 and 20 characters.
- Should not contain the following string (case sensitive): bios

Starting from BIOS version 2.11 and CIMC 3.2.10, the new BIOS password security measures are:

- BIOS password can only be set through CIMC XML API or NFVIS. It can no longer be configured in the BIOS setup menu.
- BIOS password is retained after BIOS updates and it does not have to be reconfigured after a BIOS update.
- Only an admin password can be set and user-level BIOS password can no longer be set.

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

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.

UEFI Secure Boot on ENCS 5400

The Unified Extensible Firmware Interface (UEFI) Secure Boot mode ensures that all EFI drivers and applications, ROMs or operating systems are signed and verified for authenticity and integrity before they are loaded and executed. This feature can be enabled through the GUI 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 an unsupported OS, on the next reboot, you cannot boot the device from that particular OS when you try to reboot the next time. If you try to reboot from such unsupported OS, an error is reported and recorded under System Software Events in the GUI. You must disable the UEFI secure boot option using Cisco IMC to be able to boot from the OS that does not support UEFI secure boot.

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 to take effect.

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 to take effect.

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 the installation.

To configure the UEFI virtual-mapped image as the first boot option, enter the BIOS menu using **F2** key when BIOS boots up. Use direction keys to move UEFI: Cisco CIMC-mapped image or KVM-mapped image to the top of the boot option list. For BIOS v2.10 onwards, you can also configure the UEFI boot order through CIMC GUI or CLI. For more information see, Install Cisco Enterprise NFVIS.

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.

DPDK Support on NFVIS

The Data Plane Development Kit (DPDK) is a set of data plane libraries and network interface controller drivers for fast packet processing.

DPDK Support for NFVIS 3.10.x

DPDK support is enabled only on ENCS 5400 from NFVIS 3.10.1 release. To enable DPDK use **system settings dpdk enable** command. Once DPDK is enabled it cannot be disabled. You can use **factory-default-reset all-except-images-connectivity** to disable DPDK.

To enable DPDK support:

```
configure terminal
system settings dpdk enable
commit
```

DPDK can be enabled if:

- No VMs are deployed.
- There are no other bridges created other than the default bridge which is wan-br, wan2-br or lan-br.
- The default bridges are not modified.

DPDK mode is enabled on a bridge, if the bridge is created as part of a network or bridge api without any NIC ports. NIC ports can also be added later to the bridge, if no VMs are deployed on the network associated to the bridge. If a NIC port is added to the bridge, the bridge will switch to non-dpdk mode. Once a bridge enters non-dpdk mode, it will not switch back to DPDK mode again. NFVIS supports DPDK for the interface with virtio driver only.

Note NFVIS 3.10.x release does not support **tcpdump packetcapture** command on DPDK enabled bridge.

If DPDK is enabled, all VMs deployed will have DPDK and HugePage support. The default hugepage size is 2MB. After DPDK is enabled the system reserves 512 hugepages for Openvswitch operations. Hugepages for VM are allocated dynamically. If the system is not able to allocate HugePages for a newly deployed VM, the VM will boot up in error state. Memory Fragmentation is the main reason why HugePage allocation fails. In this case a reboot can help solve the issue.

Note DPDK support is only enabled on the bridges without NIC ports.

For a system without Hyper-threading one additional core is reserved by the system and for a system with Hyper-threading two additional logical cores are reserved by the system.

NFVIS does not support changing Hyper-thread option such as disabling after DPDK is enabled with Hyper-thread. The system can be unstable if you change Hyper-thread setting after DPDK is enabled.

DPDK VM Migration for NFVIS 3.11.x

Enhancements to DPDK Support in NFVIS 3.12.x

Service Bridge DPDK Support for New Platforms

Starting from NFVIS 3.12.1 DPDK support is added for service bridges on UCS-C M5, UCS-E M3 and CSP 5000 series. Only the service bridge is supported and the default wan bridge and lan bridge do not support DPDK feature. DPDK is supported on service bridge with or without any pnic. The configuration for service bridge without any pnic is the same as ENCS platform on NFVIS 3.11.1 release, all service bridges will become dpdk-capable after DPDK is enabled. NFVIS system does not require reboot after DPDK is enabled.

DPDK and SR-IOV cannot co-exist on the pnic used for DPDK-capable bridge. The configuration for service bridge with pnic is as followed:

1. (Optional) If pnic ethX-Y is still attached to lan-br, remove it from lan-br.

```
configure terminal
bridges bridge lan-br
no port ethX-Y
commit
```

2. Remove SR-IOV networks associated with ethX-Y (usually ethX-Y-SRIOV-*).

```
configure terminal
no networks network ethX-Y-SRIOV-1
no networks network ethX-Y-SRIOV-2
...
commit
```

3. Disable ethX-Y SRIOV configuration.

```
configure terminal
pnic ethX-Y
no sriov
nnumvf 0
commit
```

4. Create service bridge with port ethX-Y.

```
configure terminal
bridges bridge service-br port ethX-Y
networks network service-net bridge service-br
commit
```

5. Enable system-wide DPDK setting.

```
configure terminal
system settings dpdk enable
commit
```



Note

- You can also enable DPDK before creating service bridge with port ethX-Y.
- Except ENCS 5400 platform, all default bridges (lan-br, wan-br, wan-br2) does not enter DPDK-mode after DPDK is enabled system-wide.
- If you do not remove SR-IOV configuration from pnic ethX-Y, and ethX-Y is a port of service bridge, this service bridge cannot enter DPDK-mode after DPDK is enabled system-wide. If you have to use pnic ethX-Y in DPDK bridge, after removing SR-IOV configuration from ethX-Y, delete old bridge and create new bridge and attach ethX-Y to the new bridge.

Migration for VMs with PNIC associated to DPDK bridge

If DPDK is enabled, all VMs deployed will have DPDK and HugePage support. The default hugepage size is 2MB. After DPDK is enabled, the system will reserve additional 512~2048 contiguous hugepages for DPDK operations. Though hugepages for VM are allocated dynamically, it will cause memory fragmentation and the main reason DPDK enabling process will fail. In this case, a system reboot might help to solve the issue.

If system can find enough hugepages for DPDK process, not only service bridges will enter DPDK-mode, all VMs' vnic attached to those bridges will be migrated from virtio to dpdk type (dpdkvhostuserclient):

```
nfvis# support ovs vsctl show | begin service-br
    Bridge service-br
        Port "vnic2"
            Interface "vnic2"
                type: dpdkvhostuserclient
                options: {vhost-server-path="/run/vhostfd/vnic2"}
        Port service-br
            Interface service-br
                type: internal

nfvis# support virsh dumpxml ROUTER | begin interface | until serial
    <interface type='network'>
      <mac address='52:54:00:ae:14:57'/>
      <source network='wan-net'/>
      <target dev='vnic1'/>
      <model type='virtio'/>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x04' function='0x0'/>
    </interface>
    <interface type='vhostuser'>
      <mac address='52:54:00:3d:ee:1a'/>
      <source type='unix' path='/run/vhostfd/vnic2' mode='server'/>
      <target dev='vnic2'/>
      <model type='virtio'/>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x05' function='0x0'/>
    </interface>
```

DPDK and Factory Reset

The three commonly-used factory reset configurations disable DPDK.

Factory-default-reset configuration	DPDK enabled	Networks kept	Deployment deleted	Registered image deleted
all	No	No	Yes	Yes
all-except-images	No	No	Yes	No
all-except-images -connectivity	No	Yes	Yes	No

Frequently Seen Failures

- Error when DPDK is enabled.
  - When enabling DPDK, lack of system memory results in an error. Ensure that you have 4 GB contiguous HugePages to enable DPDK. You can also reboot the system.
  - DPDK requires one physical core from each socket. A multi-sockets system (like UCS-C or CSP) can encounter deployments that occupy all cores on one socket and DPDK can fail to enable. Power-off some VM (resource-locking VMs likes ISRv/ASAv/vWAAS) and enable VM after DPDK is enabled successfully.
- If the system is not able to allocate HugePages for a newly deployed VM, the VM will boot up in error state.

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.
- VM export is dependant on the amount of free space available in the deployed datastore, regardless of the free space available in the destination datastore. For example, when the VM is deployed in the intdatastore (default), you should ensure that the available free space is at least twice that of the deployed VM.

To export a VM ensure that:

- 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.
```

Starting from NFVIS 4.1 release, an optional unique MAC UID support is added to VM import.

```
vmImportAction importPath <vm backup file with location> uniqueMacUid
```

Specifying the uniqueMACUid flag ensures that the imported VM is not deployed with the same UID and interface MAC addresses.

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
  ```

Feature Comparison Table for Import and Export

VM backup using vmExportAction:

Features	NFVIS 4.1.1 and Earlier Releases	NFVIS 4.2.1 Release	NFVIS 4.4.1 Release
Default file location for backup vmExportAction vmName sample exportName vmbackup exportPath <datastore>:	/data/intdatastore/vmbackup.vmbkp /mnt/extdatastore1/vmbackup.vmbkp /mnt/extdatastore2/vmbackup.vmbkp	Same	Same
Default file location for backup on USB vmExportAction vmName sample exportName backup02 exportPath usb:usb1	/mnt-usb/usb1/vmbackup.bkup	Same	Same
Check disk space before backup	Not supported	Supported	Supported
VM backup format	Full backup	Diff disk backup	Diff disk backup
Backup image and flavor	Not supported	Not supported	Supported
VM live export snapshot	Not supported	Not supported	Supported
VM Export with Selective Disk	Not supported	Not supported	Supported

VM restore using vmImportAction:

Features	NFVIS 4.1.1 and Earlier Releases	NFVIS 4.2.1 Release	NFVIS 4.4.1 Release
Default file location for backup vmImportAction importPath <datastore>:vmbackup.vmbkp	/data/intdatastore/vmbackup.vmbkp /mnt/extdatastore1/vmbackup.vmbkp /mnt/extdatastore2/vmbackup.vmbkp	Same	Same
Default file location for restore on USB vmImportAction importPath usb:usb1/vmbackup.vmbkp	/mnt-usb/usb1/vmbackup.vmbkp	Same	Same
Check disk space before backup	Not supported	Supported	Supported

Features	NFVIS 4.1.1 and Earlier Releases	NFVIS 4.2.1 Release	NFVIS 4.4.1 Release
Restore backing images and flavors	Not supported	Not supported	Supported
VM Export with Selective Disk	Not supported	Not supported	Supported

Backup and Restore NFVIS and VM Configurations

Table 3: Feature History

Feature Name	Release Information	Description
Enhancements to backup and restore of configurations	NFVIS 4.2.1	New commands are introduced to view the overall status of backup and restore process. Enhancements to backup file location and factory default options are introduced. Information on how to troubleshoot failure to restore NFVIS configurations is added.

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 and restore a single VM, use vmExportAction (for VM backup) and vmImportAction (for VM restore) APIs.
- Perform the following hostaction backup that avoids loss of VMs during hostaction restore due to insufficient disk space:
  1. Stop the functioning of the VMs that are associated with Cisco NFVIS.
  2. Perform individual image backups of the VMs using the **vmExportAction** command.
  3. Once the backup is successful, delete the VMs and the images from Cisco NFVIS.
  4. When you delete the VMs and the images, perform a host level backup with configurations-only option using the command **hostaction backup configuration-only file-path** *extdatastore2:sample-dir/sample*.
  5. Copy the backup files to a file server.
  6. Perform a factory reset using the **factory-default-reset** command.
  7. Paste the backup copied to a file server and restore the host level backup file using the **hostaction restore** *file-path extdatastore2:sample-dir/sample.bkup* command.
  8. When the restore fails due to disk storage issues, restore the configurations-only backup. When the restore is successful, restore the VMs and their images using the **vmImportAction** *importPath /mnt/extdatastore1/tiny_backup.vmbkp* command.

Restrictions for Backup and Restore on NFVIS

- The backup includes all deployed VMs and the registered images except uploaded files.
- VM restore using hostaction restore and vmImportAction requires original registered image to be on the system, on the same datastore. Missing registered image or image registered in a different datastore results in VM restore failure.

  For NFVIS 4.2 release, onlyVM restore using hostaction restore does not require original registered images on the system.
- For NFVIS 4.1 and earlier releases, NFVIS VM backup does not support differential disk backup and every backup is a full VM backup.
- For NFVIS 4.1 and earlier releases, 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.
- 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.
- NFVIS only supports backup or restore on the same release. For example, backup created in NFVIS 4.1.1 cannot be used to restore on NFVIS 4.2.1.
- Starting from NFVIS 4.5 release, secure-overlay configuration with EAP authentication is supported. However, it will be discarded if restored on a different box or on the same box after fresh-install because of encrypted password.
- Starting from NFVIS 4.5 release, single ip configuration is supported. However, it will be discarded if restored on a different box because the single ip bootstrap is tailored towards a particular box.

Feature Comparison Table for Backup and Restore

Backup using hostaction backup:

Feature	NFVIS 4.1.1 and Earlier Releases	NFVIS 4.2.1 Release
Default file location for backup	/data/intdatastore/uploads/backup.bkup /mnt/extdatastore1/uploads/backup.bkup /mnt/extdatastore2/uploads/backup.bkup	/data/intdatastore/backup.bkup /mnt/extdatastore1/backup.bkup /mnt/extdatastore2/backup.bkup
VM backup format	Full backup	Diff disk backup
Registered Image and Flavors	No	Yes
Status monitoring	No	Yes
Check disk space before backup	No	Yes

Restore using hostaction restore:

Feature	NFVIS 4.1.1 and Earlier Releases	NFVIS 4.2.1 Release
Default file location for backup	/data/intdatastore/uploads/backup.bkup /mnt/extdatastore1/uploads/backup.bkup /mnt/extdatastore2/uploads/backup.bkup	/data/intdatastore/backup.bkup /mnt/extdatastore1/backup.bkup /mnt/extdatastore2/backup.bkup
Restore images and flavors	No	Yes
Unique Mac Uid for VM	No for NFVIS 3.12.3 and earlier release	Yes
Status monitoring	No	Yes

Feature	NFVIS 4.1.1 and Earlier Releases	NFVIS 4.2.1 Release
SNMP v3 user/passphrase restore (with uniqMacUid)	v3 user/passphrase restore	If system engine ID is the same as backup, restore all v3 users. If system engine ID is different from backup, ignore v3 users restoration.
SNMP engine ID restore on different system	No	Engine ID changed to same as backup bundle

VM backup using vmExportAction:

Feature	NFVIS 4.1.1 and Earlier Releases	NFVIS 4.2.1 Release
VM backup format	Full backup	Diff disk backup

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 and save into datastore, or mounted USB storage device. Without specifying, the backup file will have *.bkup* extension.

	Backup configuration-only	Backup configuration-and-vms
Save system configurations	Yes	Yes
Save system upgrade configurations	Yes	Yes
Save system upgrade file	No	No
Save images and flavors configurations	Yes	Yes
Save image disks	No	Yes
Save deployments configurations	Yes	Yes
Save deployments disks	No	Yes

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.

Starting from NFVIS 4.2 release, use the **show hostaction backup status** command to view the status of the overall backup process and each components like system, image and flavors, vm and so on. The following is an example of the show command output after the backup process is complete:

```
nfvis# show hostaction backup status
hostaction backup status 2020-07-16T07:02:44-00:00
destination intdatastore:backup_20200704.bkup
status      BACKUP-SUCCESS
size        "2798.0 MB"
components FIREWALL
  status      BACKUP-SUCCESS
  last update 2020-07-16T07:07:38-00:00
  size        "20.49 MB"
  details     ""
components Linux
  status      BACKUP-SUCCESS
  last update 2020-07-16T07:07:36-00:00
  size        "0.01 MB"
  details     ""
components NFS
  status      BACKUP-SUCCESS
  last update 2020-07-16T07:06:44-00:00
  size        "0.01 MB"
  details     ""
components NFVIS
  status      BACKUP-SUCCESS
  last update 2020-07-16T07:02:48-00:00
  size        "0.72 MB"
  details     ""
components ROUTER
  status      BACKUP-SUCCESS
  last update 2020-07-16T07:07:35-00:00
  size        "579.89 MB"
  details     ""
components VM_Images_Flavors
  status      BACKUP-SUCCESS
  last update 2020-07-16T07:06:45-00:00
  size        "2197.73 MB"
  details     ""
nfvis#
```

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.

The restore is based on the system condition created during backup.

	Restore configuration-only	Restore configuration-and-vms
Restore system configurations	Yes	Yes

	Restore configuration-only	Restore configuration-and-vms
Restore upgrade configurations	yes, requires same upgrade files in system if the host backup was taken has such upgrade files. No, if host where backup was taken did not have any upgrade files registered. Restoree will fail.	Yes, requires same upgrade files in system if the host backup was taken has such upgrade files. No, if host where backup was taken did not have any upgrade files registered. Restore will fail.
Restore registered images and flavors	Yes, if images sources are still available (URL link is still valid, or uploaded files are still in the same locations). No, if images sources are not available (URL link is invalid, upload files are deleted or moved to new location). The restore process will fail.	Yes, restore from backup file.
Restore deployments	No	Yes, restore from backup file.

Note This means if there are upgrade files registered in the NFVIS. The backup create on this host will contain those information. If using this backup on new host or same host after factory-default-reset, the restore will fail.

	dpdk-disabled while backup	dpdk-enable while backup
dpdk-disabled while restore	Yes (system is dpdk-disabled)	Yes (system will beconverted to dpdk enabled, and VM vnic will be convereted inf needed)
dpdk-enabled while restore	No support	Yes (system is dpdk-enabled)

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
```

Starting from NFVIS 4.2 release, use the **show hostaction restore-status** command to view the status of the overall restore process and each components like system, image and flavors, vm and so on. The following is an example of the show command output after the restore process is complete:

```
nfvis# show hostaction restore-status
hostaction restore-status 2020-07-16T07:18:54-00:00
source intdatastore:backup_20200704.bkup
status RESTORE-SUCCESS
components FIREWALL.vmbkp
  status      RESTORE-SUCCESS
  last update 2020-07-16T07:26:34-00:00
  details     ""
components Linux.vmbkp
  status      RESTORE-SUCCESS
  last update 2020-07-16T07:26:03-00:00
  details     ""
components NFS.vmbkp
  status      RESTORE-SUCCESS
  last update 2020-07-16T07:25:36-00:00
  details     ""
components NFVIS
  status      RESTORE-SUCCESS
  last update 2020-07-16T07:22:03-00:00
  details     ""
components ROUTER.vmbkp
  status      RESTORE-SUCCESS
  last update 2020-07-16T07:26:55-00:00
  details     ""
components VM_Images_Flavors
  status      RESTORE-SUCCESS
  last update 2020-07-16T07:26:01-00:00
  details     ""
components intdatastore:backup_20200704.bkup
  status      VERIFICATION-SUCCESS
  last update 2020-07-16T07:18:54-00:00
  details     ""
nfvis#
```

Starting from NFVIS 4.2 release, you can backup registered images and flavors into backup package and restore these images and flavors into the system. The new system does not require a pre-registered image before system restore. If the system has existing images, flavors or deployments, the system restore erases them all and restores from its own backup. VM backup is now faster and uses less disk space compared to NFVIS 4.1 release, but it also takes up additional process, time or disk space to backup registered images and flavors.

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:

	NFVIS 4.1.x and earlier releases	NFVIS 4.2.x and later releases
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*).	• The system backup bundle is saved under */datastore/* by default.
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.	• The backup package includes original registered images.

	NFVIS 4.1.x and earlier releases	NFVIS 4.2.x and later releases

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`
		• Save backup bundle in local */intdatastore/* and not in */intdatastore/uploads/* or copy backup bundle from */datastore/uploads/* to */datastore/*

You can copy backup file to intdatastore/ if there is sufficient storage space. If the backup is larger than free disk space in intdatastore/, you can copy to a remote server like scp or NFVIS web portal.

The following table lists the data erased and retained upon using NFVIS factory default reset options:

	Factory-default-reset all	Factory-default-reset all-except-images	Factory-default-reset all-except-images-connectivity
files under intdatastore	Retain	Retain	Retain
files under intdatastore/uploads/	Delete	Delete	Delete

	Factory-default-reset all	Factory-default-reset all-except-images	Factory-default-reset all-except-images-connectivity
files under extdatastore${1,2}	Delete	Retain	Retain
files under extdatastore${1,2}/uploads/	Delete	Delete	Delete
files under USB	Retain	Retain	Retain
files under NFS mounted datastore	Retain	Retain	Retain
Deployments	Delete	Delete	Delete
Registered Images and Flavors	Delete	Retain	Retain

Failure to Restore

NFVIS configurations fails to restore if:

- There is no sufficient disk space. Restore requires temporary disk space to save un-compressed files. You can move, copy or upload the backup file to a larger datastore and run system restore.

```
nfvis# show hostaction restore-status
hostaction restore-status 2020-07-16T21:29:08-00:00
source intdatastore:encs07-configVms-dpdk-2020-07101600.bkup
status RESTORE-ERROR
components intdatastore:encs07-configVms-dpdk-2020-07101600.bkup
  status      VERIFICATION-ERROR
  last update 2020-07-16T21:49:18-00:00
  details     "Backup package could not be inflated. No space left on device"
nfvis#
```

- The application communication fails. You can see this error after the first restore attempt has failed, and when you try to restore for the second time. You can reboot NFVIS before you attempt restore again.

```
nfvis# hostaction restore file-path extdatastore2:backup_20200704.bkup
Error: application communication failure
```

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). To disable SR-IOV on a PNIC, set the SR-IOV value to 0. To enable SR-IOV on a PNIC, set the SR-IOV value between 1 and the maximum number of virtual functions (maxvfs) supported on that PNIC. 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. The existing fresh installation behavior has not changed. Each PNIC has a number of VFs and SR-IOV networks created by default. You can use CLI, API, or the GUI to enable and disable SR-IOV on a PNIC and to create and delete SR-IOV networks.

Note The number of SR-IOV networks, numvfs or inusevfs, created per PNIC on fresh installation of NFVIS depends on the link speed of that particular pnic.

Restrictions and Limitations

- The supported platforms are CSP-2100, CSP-5228, CSP-5436, CSP-5444 (Beta), Cisco Catalyst 8200 UCPE, 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

- NFVIS release 3.12.1 supports Virtual Ethernet Bridge (VEB) in switch mode only.
- 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

To disable SR-IOV on a PNIC, ensure that:

- All SR-IOV networks on a PNIC are deleted.
- The PNIC is not 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, ensure that:

- The PNIC supports SR-IOV.
- The numvfs field is populated with a value that is less than the maximum number of virtual functions (maxvfs) supported on that PNIC.
- The PNIC is not attached to a bridge.

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

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

Create SR-IOV Networks

To create SR-IOV networks, the PNIC must have SR-IOV enabled and configured with numvfs. The SR-IOV network name must have the following format: <pnic_name>-SRIOV-<num> where <pnic_name> is a valid PNIC name and <num> is a value that is greater than 0 and less than the number of VFs (numvfs).

To create an SR-IOV network in trunk mode:

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

To create an SR-IOV network in access mode:

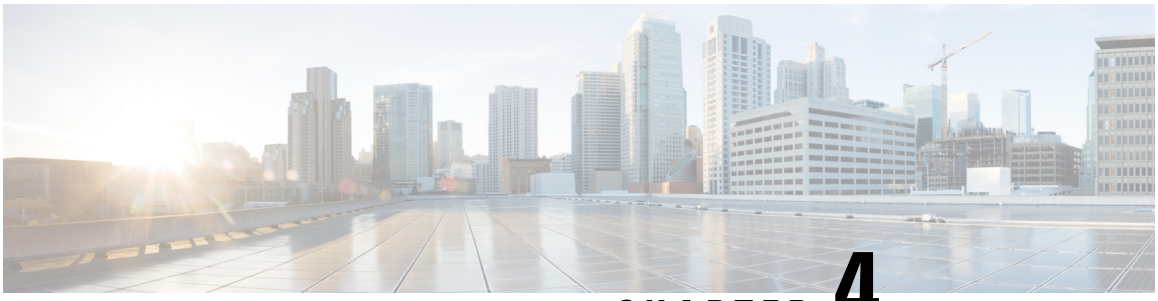
```
configure terminal
networks network eth0-1-SRIOV-1 sriov true trunk false vlan 30
commit
```

Delete SR-IOV Networks

To delete an SR-IOV network, ensure that no VMs are attached to the network.

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

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



CHAPTER 4

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

Note NFVIS SD-Branch Zero Touch Provisioning (ZTP) PnP redirect fails if the WAN DHCP IP address is in the same subnet as management IP address 192.168.1.0/24. For a successful PnP redirect, you can change the DHCP pool to a different subnet or manually change the management IP address to a different subnet.

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, on page 62
- Configuring PnP Discovery Methods, on page 63
- PnP Action, on page 66

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, starts 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:

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

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.

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

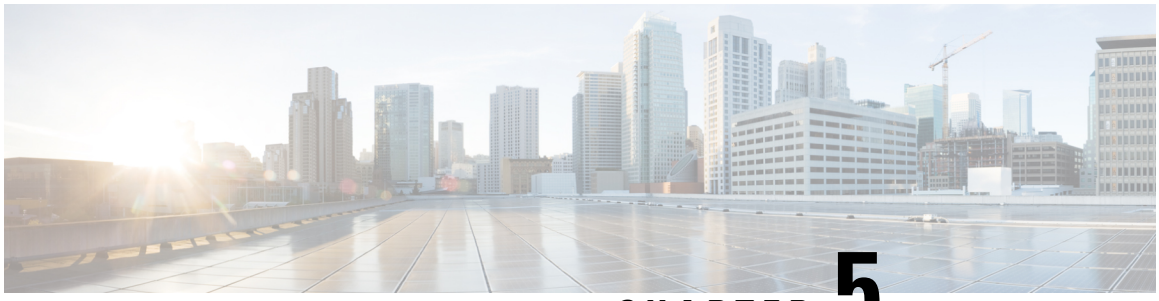
PnP Server APIs	PnP Server Commands
• /api/config/pnp • /api/config/pnp?deep	• pnp static ip-address • pnp automatic • show pnp

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



CHAPTER 5

Overview of VM Life Cycle Management

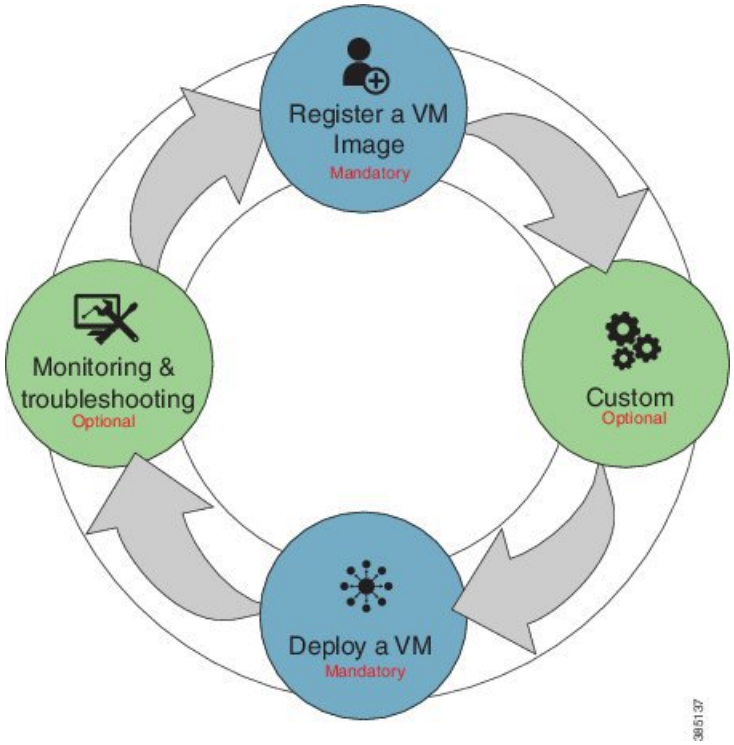
VM life cycle management refers to the entire process of registering, deploying, updating, monitoring VMs, and getting them service chained as per your requirements. You can perform these tasks and more using a set of REST APIs or NETCONF commands or the Cisco Enterprise NFVIS portal.

- Workflow of VM Life Cycle Management, on page 67
- Uploading VM Images to an NFVIS Server, on page 69
- VM Bootstrap Configuration Options with a VM Deployment, on page 70
- OpenStack Configuration Drive Support for Third Party VMs, on page 71
- Performing Resource Verification, on page 72
- Configuring Management IP Subnet, on page 73
- VM States, on page 73

Workflow of VM Life Cycle Management

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

Figure 5: VM Life Cycle Management



1. **Register a VM Image—**To create a VM instance, you must register the source image in the NFVIS image repository. Registering the image is a one-time activity. The source image is expected to be hosted on a remote server. In the Registration API you specify the URL on the remote server, where the file is located. When invoked, device pulls the API info, completes the file transfer and registration. Once registered, you can create one or more VM instances. Remote servers supported for hosting the source images (.tar.gz files) are HTTP/HTTPS servers. Starting with NFVIS 4.12.1 release, remote servers can also be FTP servers or SCP servers.

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

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.

OpenStack Configuration Drive Support for Third Party VMs

To enable staging of bootstrap configuration files at the time of a third party VM deployment as per OpenStack standards, the following cloud init format is supported:

```
openstack/content
openstack/content/0000
openstack/content/0001
openstack/latest/meta_data.json
```

In the above sample, the "0000" and "0001" files are the actual bootstrap files from the deployment payload. A third party VM can use the init file to fetch its configuration files.

The following metadata file is used to provide the file path on the configuration drive and reference to the actual bootstrap configuration files.

```
{
    "files": [
        {
            "content_path": "/content/0000",
            "path": "/config/day-0.txt"
        },
        {
            "content_path": "/content/0001",
            "path": "/sample/path/iosxe_config.txt"
        }
    ]
}
```

With this implemenation, two copies of the same bootstrap configuration file will be present on the virtual CD-ROM package. The first version at the root (iosxe_config.txt) and the second inside the "openstack/content" folder.

The admin will also have to sepcify the bootstrap configuartion file in the image properties file before packaging the VM.

Example for the Bootstrap Configuartion File in the Image Properties File

```
--optimize=OPTIMIZE [REQUIRED] optimized VM: --optimize=true/false;
 --root_file_disk_bus=ROOT_FILE_DISK_BUS root disk file type:
 --root_file_disk_bus=virtio/ide; default is virtio
 --virtual_interface_model=VIRTUAL_INTERFACE_MODEL
 --virtual_interface_model=rtl8139; default is none
 --thick_disk_provisioning=THICK_DISK_PROVISIONING
 --thick_disk_provisioning=true; default is false
 --bootstrap_cloud_init_bus_type=BOOTSTRAP_CLOUD_INIT_BUS_TYPE
 --bootstrap_cloud_init_bus_type=virtio; default is ide
 --bootstrap_cloud_init_drive_type=BOOTSTRAP_CLOUD_INIT_DRIVE_TYPE
 --bootstrap_cloud_init_drive_type=disk; default is cdrom
 --bootstrap=BOOTSTRAP bootstrap file/s for VM (two parameters required in the format of
dst:src; dst filename including path has to match exactly to what the VM expects;
 upto 20 bootstrap files are accepted.)
 examples:
 --bootstrap ovf-env.xml:file1,ios-xe.txt:file2 for ISRv; both files get mounted at the
root level on the VM.
 --bootstrap day0-config:filename1 for ASAv
 --bootstrap
 /:bootstrap.xml,/license/lic.txt:license.txt
```

```
 bootstrap.xml get mounted as bootstrap.xml at root, and license.txt get mounted as
/license/lic.txt.
```

Note If any of the strings in the configuration file has wild characters, wrap the string with this #[[ ]]# so that the token/key replacement engine does not consider wild characters as key or token, and looks for key value pairs to replace during a VM deployment.

For details on the OpenStack standards, visit *http://docs.openstack.org*.

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
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.

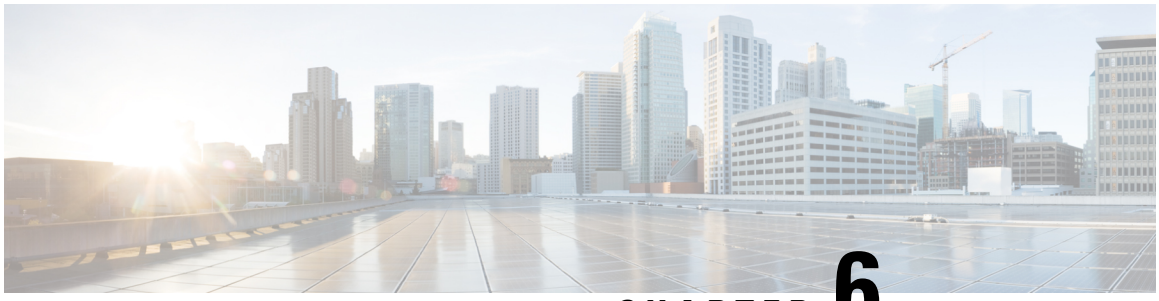
To delete an existing subnet use **no vm_lifecycle networks network int-mgmt-net subnet int-mgmt-net-subnet** command.

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 States

VM States	Description
VM_UNDEF_STATE	The initial state of VM or VNF before deployment of this VM.
VM_DEPLOYING_STATE	VM or VNF is being deployed on to the NFVIS.
VM_MONITOR_UNSET_STATE	VM or VNF is deployed in the 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 on the VM or VNFs 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_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 per the monitor or kpi metric.
VM_UNDEPLOYING_STATE	VM or VNF is being undeployed or terminated.
VM_ERROR_STATE	VM or VNF will be in error state if deployment or any other operation is failed.



CHAPTER 6

VM Deployment Scenarios

This chapter provides details on the following deployment scenarios using REST APIs. As an example, the Cisco ENCS is used to illustrate these scenarios.

- Single VM deployment
- Service chaining with two VMs
- Service chaining of multiple VMs with Windows or Linux servers

  The following VM images are used to explain the deployment scenarios:

  - Cisco Integrated Services Router (ISRv) —isrv-03.16.02
  - Cisco Adaptive Security Virtual Appliance (ASAv)— asav951-201
  - Linux server—ubuntu-14.04.3-server-amd64-disk1

- Registering VM Images, on page 75
- Service Chaining of VMs, on page 79

Registering VM Images

You must register all VM images before deploying them.



Note Register all the VM images required for the VM deployment depending on the topology. A VM image registration is done only once per VM image. You can perform multiple VM deployments using the registered VM image.

To register a Cisco ISRv image:

1. Set up the http/https server to host the VM image, or upload the image to the NFVIS server using the local portal or the **scp** command.
2. Register the Cisco ISRv image using the following API method:

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

```
POST https://<NFVIS_IP>/api/config/vm_lifecycle/images -d
'<image><name>isrv-k9.16.03.01</name><src>http://filename_with_full-path-of
the-file/isrv-universalk9.16.03.01.tar.gz</src></image>'
```

3. Verify the image status using the following API method:

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X
GET
https://<NFVIS_IP>/api/operational/vm_lifecycle/opdata/images/image/isrv-9.16.03.01?deep
```

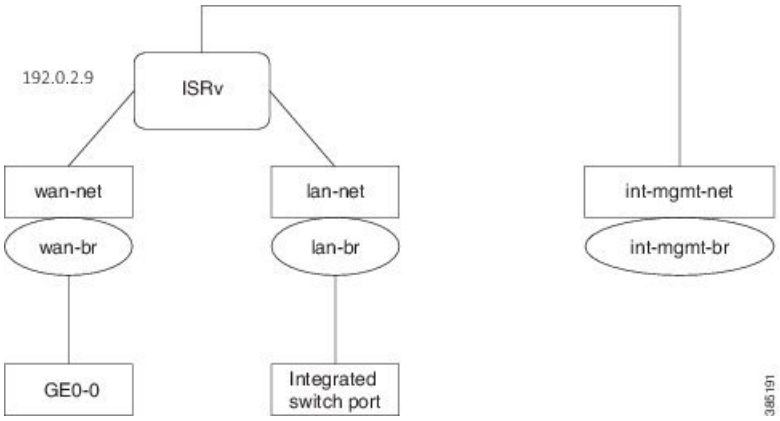
4. Now, repeat Steps 1 to 3 to register the Cisco ASAv and Linux server images. Ensure that you provide the exact image name and source file location when running the API commands.

Note You can run API commands from any console/server that can reach Cisco Enterprise NFVIS.

Single VM Deployment

In this example, a Cisco ISRv image with three network interfaces is deployed. The following diagram illustrates the deployment topology:

Figure 6: Single VM Deployment



Steps for Deploying a VM

To deploy a Cisco ISRv image:

1. Verify that all networks required for your deployment are configured.

```
curl -k -v -u admin:admin -H content-type:application/vnd.yang.data+xml -X
GET https://<NFVIS_IP>/api/config/networks?deep
```

2. Before deploying the VM, you can perform a resource check to ensure that you have sufficient resources for the deployment.

```
curl -k -v -u "admin:admin" -X GET
https://<NFVIS_IP>/api/operational/resources/precheck/vnf/newvnf,isrv-small,true
?deep
```

3. Deploy the Cisco ISRv 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://<NFVIS_IP>/api/config/vm_lifecycle/tenants/tenant/admin/deployments --data
<deployment>
    <name>ISR</name>
    <vm_group>
        <name>ISR</name>
        <image>isrv-universalk9.16.03.01/image>
        <bootup_time>600</bootup_time>
        <recovery_wait_time>0</recovery_wait_time>
        <recovery_policy>
            <action_on_recovery>REBOOT_ONLY</action_on_recovery>
        </recovery_policy>
        <flavor>isrv-small</flavor>
        <interfaces>
            <interface>
                <nicid>0</nicid>
                <network>int-mgmt-net</network>
                <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>
            <interface>
                <nicid>1</nicid>
                <network>lan-net</network>
                <ip_address>209.165.201.0</ip_address>
            </interface>
            <interface>
                <nicid>2</nicid>
                <network>wan-net</network>
                <ip_address>209.165.201.1</ip_address>
            </interface>
        </interfaces>
        <scaling>
            <min_active>1</min_active>
            <max_active>1</max_active>
        </scaling>
        <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>security</val>
                </variable>
                <variable>
                    <name>ngio</name>
                    <val>enable</val>
                </variable>
            </configuration>
        </config_data>
    </vm_group>
</deployment>
```

Note If two VM's are connected to the same physical interface, one over SRIOV and another over virtio net from a bridge connected to the packet filter, you cannot ping between the two VM's between these interfaces. Use SRIOV or virtio net on both the VM's to connect to the packet filter over a bridge.

4. Verify the deployment 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://NFVIS_IP/api/operational/vm_lifecycle/opdata/tenants/tenant/admin/deployments/ISR,-,-?deep
```

Note To enable NIM support on a Cisco ISRv running on Cisco ENCS, you must use the following variable in the ISRv deployment payload.

```
<variable>
     <name>ngio</name>
     <val>enable</val>
</variable>
```

Service Chaining of VMs

Service chaining here refers to a set of network services in the form of VMs using an intermediate network. Cisco Enterprise NFVIS supports service chaining of two or more VMs eliminating the need of dedicated hardware devices for different types of network services.

To service chain traffic between two or more VMs, you will have to create the following:

- Bridge—For example, you can create a new bridge called sc-br.
- Network—For example, you can create a new network called sc-net.
- Launch VM1 and VM2 with an interface from each VM to the service chain network (sc-net).

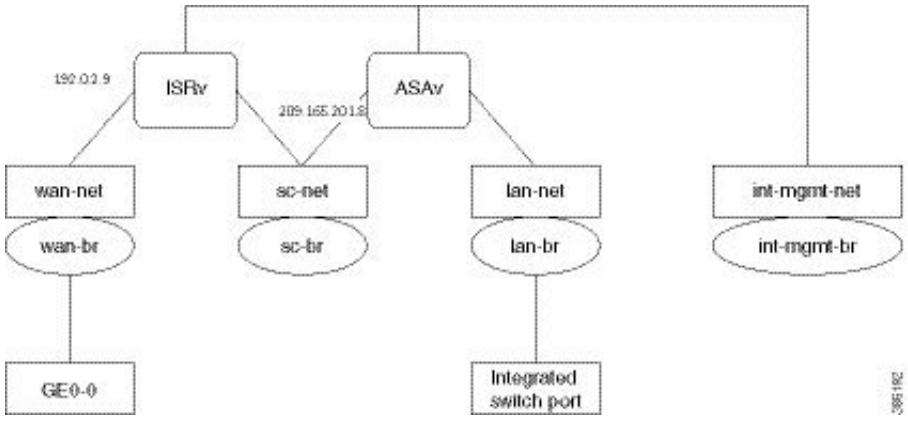
For more details on how to configure service chaining using APIs, see the following topics:

- Service Chaining with two VM Images, on page 79
- Service Chaining of Multiple VMs with Windows or Linux Servers, on page 80

Service Chaining with two VM Images

In this example, a Cisco ISRv VM and a Cisco ASAv VM are service chained. For that, you will have to deploy both VMs.

Figure 7: Service Chaining with two VM Images



Steps for Service Chaining with Two VM Images

1. Create a new bridge for service chaining.

```
curl -k -v -u admin:admin -H content-type:application/vnd.yang.data+xml -X POST
https://<NFVIS_IP>/api/config/bridges --data
'<bridge><name>sc-br</name></bridge>'
```

2. Create a new network for service chaining, and attach the bridge to the network.

```
curl -k -v -u admin:admin -H content-type:application/vnd.yang.data+xml -X POST
https://<NFVIS_IP>/api/config/networks --data
```

```
'<network><name>sc-net</name><bridge>sc-br</bridge> </network>'
```

3. Verify that all bridges and networks are configured.
4. Deploy the Cisco ISRv VM, and verify the deployment status.
5. Deploy the cisco ASAv VM, and verify the deployment status.

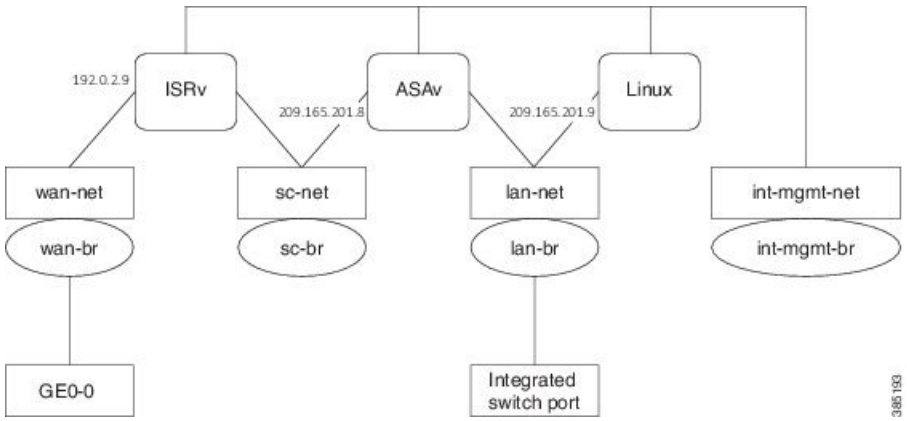
   See Steps for Deploying a VM , on page 76 for API command details for Steps 3 to 5.

Service Chaining of Multiple VMs with Windows or Linux Servers

In this example, multiple VMs will be service chained. Cisco ISRv and Cisco ASAv VMs can be deployed as explained in Service Chaining with two VM Images, on page 79.

This section covers Linux server deployment (Windows 2012 server can also be deployed using the same steps.)

Figure 8: Service Chaining of Multiple VMs with Windows or Linux Servers

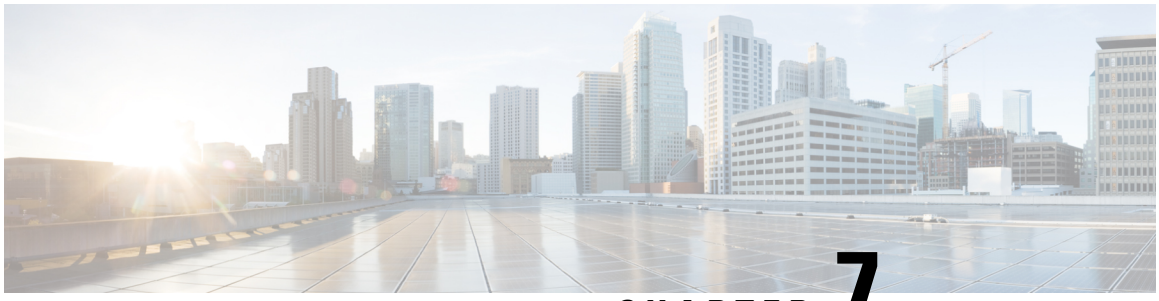


Steps for Service Chaining of Multiple VMs with Windows or Linux Servers

1. Create networks and bridges as required.

   See Steps 1 and 2 in Steps for Service Chaining with Two VM Images, on page 79 for details on creating networks and bridges.
2. Deploy Cisco ISRv and Cisco ASAv, and verify their deployment status.
3. Deploy the Linux server VM.
4. Verify the server deployment status.

   See the Steps for Deploying a VM , on page 76 for API command details for Steps 2 to 4.



CHAPTER 7

SPAN Session or Port Mirroring

- About SPAN Sessions, on page 81
- Configuring SPAN Sessions, on page 81
- Configuration Examples for SPAN Session Scenarios, on page 83

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:
  - You can send the mirrored traffic to interfaces of Cisco ENCS switch port when you use a physical interface.
  - 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.



Note Use VM Group Name for **source vm-vnic** and **destination vm-vnic** commands. Use the **show vm_lifecycle deployments all | include Group** command to get Group Name.

```
ENCS# show vm_lifecycle deployments all | include Group
VM Group Name : deployment-ASAv
VM Group Name : deployment-ISRv
VM Group Name : deployment-Linux
ENCS#
```

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
```

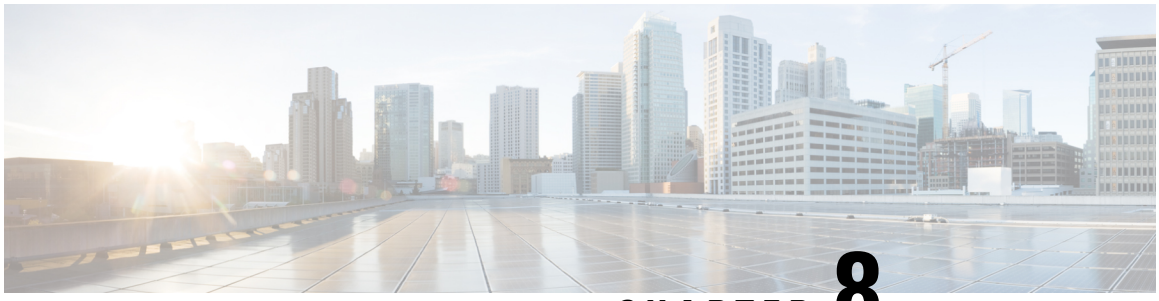
Example: SPAN Session Traffic on a VLAN

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
```



CHAPTER 8

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>

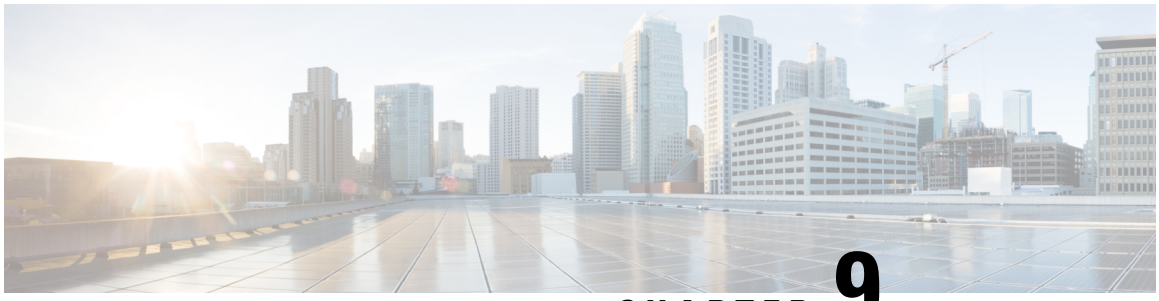
User Name:cisco
Password:**********.   <this is the password you get from step 1 above>
```

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 9

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, on page 89
- Standard VM Image Packaging, on page 98
- Appendix, on page 98

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.
- tool-usage-examples.txt—This file contains examples on how to use the image packaging utility to package a VM image.

- scaffold_template.cfg— This is a json format configuration file that specifies the meta-data for a VM to be packaged.

Usage

VM Image Packaging Tool with a UI

Table 4: Feature History Table

Feature	Description
VM Image Packaging Tool with a UI	Starting from Cisco NFVIS Release 4.9, a new VM packaging tool is introduced. This VM image packaging tool helps in generating a full VM package. In addition, scaffolding and repacking can be performed using the UI elements in the tool. **Note** The VM image packaging tool is not a part of Cisco NFVIS. The tool needs to be downloaded and installed in your system.

Download VM Image Packaging Tool

Use the following steps to download and install the VM image packaging tool:

1. Navigate to the Software Download page of NFVIS Release 4.9.1.
2. Click the download icon next to **VM Packaging tool for NFVIS** file.
3. Unzip the **CISCO-NFVIS-vmpackagingtool-4.9.1.tar** file.
4. The unzipped folder contains **nfvpt_ui_README.txt** and **nfvpt_ui.tar** files.
5. Follow the instructions in the **nfvpt_ui_README.txt** file to install the tool and perform VM image packaging.

VM Packaging using the CLI

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.

Starting from NFVIS 4.6 and later releases, use the following **help** command:

```
nfvpt.py --help

Cisco NFVIS Packaging Tool

usage: nfvpt.py packaging-v2 [-h] [--config CONFIG] [-o PACKAGE_FILENAME]
                             [--log LOG] [-d PACKAGE_OUTPUT_DIR]
                             [--legacy_help LEGACY_HELP]

Version: 4.6+ Cisco NFVIS: VNF image packaging utility

optional arguments:
```

```
  -h, --help            show this help message and exit
  --config CONFIG       [REQUIRED] config file template for creation of the
                        package.
  -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
  --log LOG             [REQUIRED] file name for logfiles
  -d PACKAGE_OUTPUT_DIR, --package_output_dir PACKAGE_OUTPUT_DIR
                        destination path where package is required
  --legacy_help LEGACY_HELP
                        show the help message for the options available in
                        legacy packaging version and exit
```



Note You can also use the **nfvpt.py --legacy_help** command to get information about the previous version of the tool command help section.

For NFVIS 4.5 and earlier releases, use the following **help** command:

```
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*.

Parameter	Mandatory/Optional	Description
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.
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.

Parameter	Mandatory/Optional	Description
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.
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 *tool-usage-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
```

Usage examples for NFVIS 4.6 and later releases:

Example 1: Usage for configuration template based packaging

```
nfvpt.py packaging-v2 --config scaffold_template.cfg --log logger -o example.tar.gz -d dest
```

Example 2: Usage for adding the certifiate file to the Day 0 file (only supported for C8000v and vEdge)

```
nfvpt.py --modify_package repackage --file_path /usr/data/example1.tar.gz --cert
BranchCert.pem --package_output example2.tar.gz --mod_version X.X --root_image
/Users/data/example.qcow2
```

Example 3: Usage for adding and replacing the qcow2 file and modifying the version

```
nfvpt.py --modify_package repackage --file_path /usr/data/example1.tar.gz --package_output
 example2.tar.gz --mod_version X.X --root_image /Users/data/example.qcow2
```

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 5: 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 6: 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 7: 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"/>
```

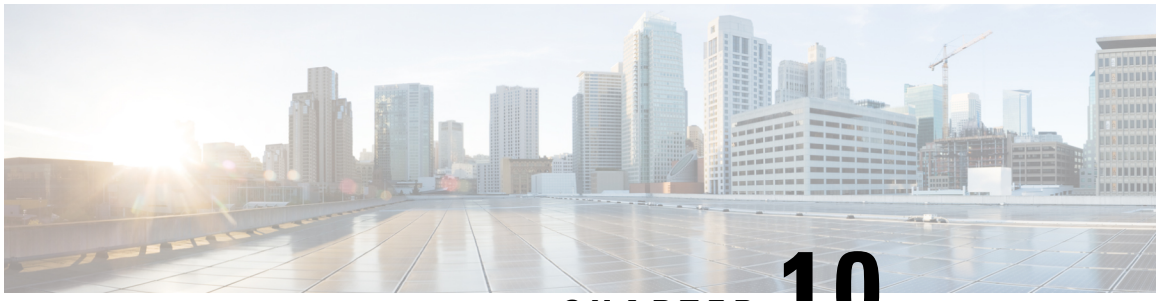
```
    <Property oe:key="com.cisco.isrv.ios-config-0009" oe:value="ip address
${NICID_0_IP_ADDRESS} ${NICID_0_NETMASK}"/>
    <Property oe:key="com.cisco.isrv.ios-config-0010" oe:value="no shut"/>
    <Property oe:key="com.cisco.isrv.ios-config-0011" oe:value="exit"/>
    <Property oe:key="com.cisco.isrv.ios-config-0012"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



CHAPTER 10

Upgrading Cisco Enterprise NFVIS

The Cisco Enterprise NFVIS upgrade image is available as a `.nfvispkg` file. Currently, downgrade is not supported. All RPM packages in the Cisco Enterprise NFVIS upgrade image are signed to ensure cryptographic integrity and authenticity. In addition, all RPM packages are verified during Cisco Enterprise NFVIS upgrade. For more information about the Image Signing and Verification feature, see Image Signing and Verification, on page 6.

Ensure that you copy the image to the Cisco Enterprise NFVIS server before starting the upgrade process. Always specify the exact path of the image when registering the image. Use the **scp** command to copy the upgrade image from a remote server to your Cisco Enterprise NFVIS server. When using the **scp** command, you must copy the image to the "`/data/intdatastore/uploads`" folder on the Cisco Enterprise NFVIS server. The following is an example on how to use the **scp** command to copy the upgrade image:

```
scp -P 22222 nfvis-351.nfvispkg admin@192.0.2.9:
/data/intdatastore/uploads/nfvis-351.nfvispkg
```

Alternatively, you can upload the image to the Cisco Enterprise NFVIS server using the **System Upgrade** option from the Cisco Enterprise NFVIS portal.

Note When the NFVIS upgrade is in progress, ensure that the system is not powered off. If the system is powered off during the NFVIS upgrade process, the system can become inoperable and you need to reinstall the system.

The upgrade process comprises two tasks:

- Registering the image using the **system upgrade image-name** command.
- Upgrading the image using the **system upgrade apply-image** command.

Registering an Image

To register an image:

```
configure terminal
system upgrade image-name nfvis-351.nfvispkg location /data/intdatastore/uploads
commit
```

Note You must verify the image registration status before upgrading the image using the **system upgrade apply-image** command. The package status must be valid for the registered image.

Verifying the Image Registration

Use the **show system upgrade reg-info** command in the privileged EXEC mode to verify the image registration.

```
nfvis# show system upgrade reg-info
PACKAGE
NAME              LOCATION                                    VERSION         STATUS UPLOAD DATE
-------------------------------------------------------------------------------------------------------------------
nfvis-351.nfvispkg /data/upgrade/register/nfvis-351.nfvispkg 3.6.1-722 Valid
2017-04-25T10:29:58.052347-00:00
```

Upgrading the Registered Image

To upgrade the registered image:

```
configure terminal
system upgrade apply-image nfvis-351.nfvispkg scheduled-time 5
commit
```

Verifying the Upgrade Status

Use the **show system upgrade apply-image** command in the privileged EXEC mode

```
nfvis# show system upgrade apply-image
UPGRADE
NAME   STATUS     FROM     UPGRADE TO
-------------------------------------------------------------------------------------------------------
nfvis-351.nfvispkg SUCCESS 3.5.0 3.5.1
```

The only upgrade supported when BIOS secured boot (UEFI mode) is enabled on ENCS 5400 platform is:

NFVIS 3.8.1 + BIOS 2.5(legacy) --> NFVIS 3.9.1 + BIOS 2.6(legacy)

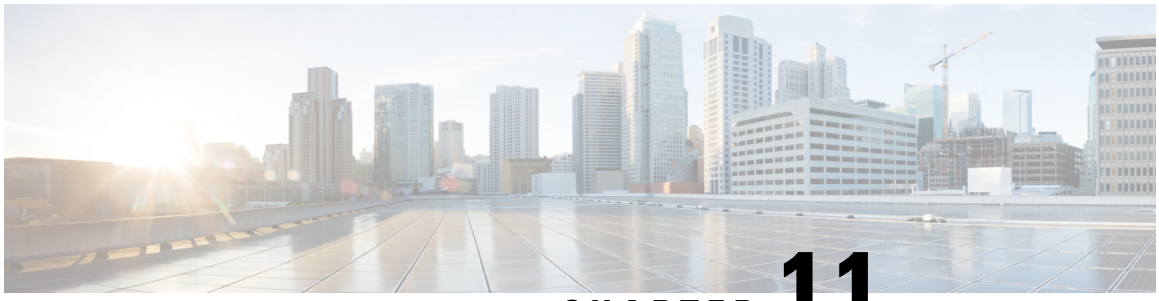
The following upgrade requires re-installation:

NFVIS 3.8.1 + BIOS 2.5(legacy) --> NFVIS 3.9.1 + BIOS 2.6(UEFI)

NFVIS 3.9.1 + BIOS 2.6(legacy) --> NFVIS 3.9.1 + BIOS 2.6(UEFI)

Upgrade APIs and Commands

Upgrade APIs	Upgrade Commands
• /api/config/system/upgrade • /api/config/system/upgrade/image-name • /api/config/system/upgrade/reg-info • /api/config/system/upgrade/apply-image	• system upgrade image-name • system upgrade apply-image • show system upgrade reg-info • show system upgrade apply-image



CHAPTER 11

Configuring vBranch High Availability

The vbranch high availability (HA) solution is a box-to-box HA. It is similar to the traditional branch, which uses physical boxes for routing and other services. This solution uses the Hot Standby Router Protocol (HSRP), a default gateway redundancy (or a first hop redundancy), which allows the network to recover from the failure of the device acting as the default gateway for the LAN side end points (devices). The routing protocols are configured to converge the traffic on the WAN side, when there are failures. So, this solution uses HSRP to provide redundancy for the branch connectivity on the LAN side. The Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF) routing protocols, and Embedded Event manager (EEM) scripts are configured to converge on the WAN side. The following section explains the redundancy solutions for the branch, with each ENCS having separate active WAN link.

Note You can use this recommended HA design as is, or modify as per the field requirement.

- Prerequisites for vBranch HA, on page 113
- vBranch HA Design and Topology, on page 114
- Enable Virtual NIC Failure Detection with Track Feature , on page 114
- Isolating LAN and Transit Link Traffic for vBranch HA, on page 116
- Packet Flow for vBranch HA, on page 118
- Configuration Examples for vBranch HA, on page 119
- Cisco ENCS Failure Points, on page 122

Prerequisites for vBranch HA

- Cisco ISRv must run HSRP on the LAN facing interface.
- 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.
- A transit link, which is L3 routed, is configured between the Cisco ENCS devices. Since the LAN HSRP makes only one device active, the transit link is used to forward traffic This link is used to forward traffic from the standby ENCS WAN to LAN or LAN to WAN. This link can be back-to-back connected on the ENCS internal switch ports.

- VMs and VNFs on both ENCS devices must be configured identical.

vBranch HA Design and Topology

Physical Devices Connection

Each Cisco ENCS has a WAN traffic connected to the Gigabit Ethernet interface, GE0-0, in this dual-WAN topology.

There are two Cisco ENCS devices namely ENCS1 and ENCS2. There is an external switch connecting one of the LAN ports from each Cisco ENCS. There is a back-to-back connection between ENCS1 and ENCS2 connecting one of the LAN ports from each Cisco ENCS. The WAN port from each Cisco ENCS is connected to the service providers network.

ISRv1 on ENCS1 and ISRv2 on ENCS2 are responsible for handling packets from LAN to WAN and WAN to LAN. If the WAN connection goes down or if the ISRv1 becomes unavailable, fast converging routing protocols, such as EIGRP and OSPF, can respond within seconds so that ISRv2 is prepared to transfer packets.

VM and Service Chain Network Connection

The Cisco ISRv should be created with an additional vNIC mapped to the transit link between two Cisco ENCS devices, apart from the regular WAN and LAN or service net links. The Cisco ISRv on both ENCS should have identical resource configurations (vNICs, vCPU, memory, etc.) and feature configurations.

Each Cisco ENCS is running an instance of service VNFs (for example, Cisco ASAv and Cisco vWAAS), and should have the identical service chain VNFs configured on both Cisco ENCS devices. Service VNFs should also have same features configured on both Cisco ENCS devices. The traffic goes through the service VNFs on the active Cisco ENCS only, even though both Cisco ENCS devices are actively forwarding on the WAN link. On a failover, the traffic will go over the service VNFs on the newly active ENCS (ENCS2).

This HA solution requires a transit link configured between two Cisco ENCS devices. One of the LAN ports from each of the Cisco ENCS can be connected back to back. This transit link port should be extended to the Cisco ISRv.

Enable Virtual NIC Failure Detection with Track Feature

You can enable the Track feature to detect virtual NIC failure in the following two scenarios:

- When the underlying physical link fails, the HSRP or routing protocols cannot detect the failure—This is because the line protocol does not go down when the underlying physical link fails if the Cisco ISRv is using a virtual NIC.
- With EEM scripts unconfigured, when the underlying physical link fails, the virtual NIC line protocol does not go down. In this case the routing protocol does not withdraw the routes.

Configuration Example for the Track Feature with Scenario 1 (HSRP)

In the virtual environment, you can enable higher protocols like HSRP to take action when the link failure happens. One way to achieve this is by configuring the Track feature on some object (ICMP ping) in Cisco IOS XE.

In Cisco ISRv, if the LAN interface where HSRP is running is a virtual NIC, then you can configure the track object to ping some device on the LAN segment, and monitor the connection failures. So, when the track object is down due to some connection failure, you can configure an action as to shut down the HSRP group, so that the peer will take over the active role making the default Gatway IP active. Without this track object, both Cisco ENCS devices will become active getting into a split-brain scenario.

The following example shows how to configure the track object on the active ISRv1, and monitor the connection failures by pinging the device IP in the network.



Note The Cisco ISRv should have AX license to configure the IP SLA.

```
track 1 ip sla 1 reachability
ip sla 1
 icmp-echo 192.0.2.1 source-ip 198.51.100.1
 frequency 5
ip sla schedule 1 life forever start-time now
!
track 5 ip sla 5 reachability
ip sla 5
 icmp-echo 192.0.2.2 source-ip 198.51.100.2
 frequency 5
ip sla schedule 5 life forever start-time now
!
```

The following output shows that the Track 1 reachability is failed, and Track 5 is up.

```
device1# show track
Track 1
  IP SLA 1 reachability
  Reachability is Down
    11 changes, last change 00:01:22
  Latest operation return code: Timeout
  Tracked by:
    HSRP GigabitEthernet3 25
Track 5
  IP SLA 5 reachability
  Reachability is Up
    4 changes, last change 00:02:32
  Latest operation return code: OK
  Latest RTT (millisecs) 1
  Tracked by:
    HSRP GigabitEthernet3 25
ISRv1#
```

The following example shows how to configure the Track object to monitor the line protocol state of the interface:

```
track 2 interface GigabitEthernet2 line-protocol
```

The following output shows that the line protocol state is down:

```
device# show track
Track 2
  Interface GigabitEthernet2 line-protocol
  Line protocol is Down ((hw down))
    8 changes, last change 00:01:25
```

```
  Tracked by:
    HSRP GigabitEthernet3 25
```

Configuration Example for the Track Feature with Scenario 2 (EEM Scripts)

With EEM scripts unconfigured, when an underlying link fails, the virtual NIC line protocol does not go down. This causes the problem as the routing protocol will not withdraw the routes. You can configure a Track object (can use the same object defined for HSRP above) to detect the failure. When the failure happens, the active Cisco ISRv has to withdraw the routes or network, so that the WAN link does not receive any traffic. One way to withdraw the routes is configure the EEM script, and delete the network from EIGRP.

The following example shows how to configure the EEM scripts, and remove the network from EIGRP:

```
track 5 ip sla 5 reachability
!
ip sla 5
 icmp-echo 192.0.2.1 source-ip 192.0.2.18
 frequency 5
ip sla schedule 5 life forever start-time now
!
event manager applet noshut_int
 event track 5 state up
 action 1.1 cli command "enable"
 action 1.2 cli command "config t"
 action 1.3 cli command "router eigrp 10"
 action 1.4 cli command "network 192.0.2.1 0.0.0.255"
 action 1.5 cli command "end"
event manager applet shut_int
 event track 5 state down
 action 1.1 cli command "enable"
 action 1.2 cli command "config t"
 action 1.3 cli command "router eigrp 10"
 action 1.4 cli command "no network 192.0.2.1 0.0.0.255"
 action 1.5 cli command "end"
```

In the virtual environment HSRP, make sure to use standby use-bia.

The following configuration example shows how to use the Track object (Track 5) to shut down HSRP group in ISRv1, when reachability is down for Track 5:

```
interface GigabitEthernet4
 description Service-NET-Virtio
 ip address 192.0.2.1 255.255.255.0
 standby use-bia
 standby 25 ip 192.0.2.22
 standby 25 timers 1 5
 standby 25 priority 105
 standby 25 preempt
 standby 25 track 5 shutdown
```

Isolating LAN and Transit Link Traffic for vBranch HA

LAN traffic and transit link traffic shall be isolated by configuring different VLANs for each traffic since both links are connected to the same ENCS internal switch. If you do not isolate these traffic, both LAN traffic and transit link will flow through the same internal switch on the Cisco ENCS.

The following Cisco ENCS switch configuration example shows how to isolate traffic. In this example, the Cisco ISRv is configured to send HSRP traffic as an untag and transit traffic in VLAN 46. So, to isolate HSRP traffic and transit traffic on the internal switch, the Gigabit Ethernet interface 1/0 is connected to a LAN network and Gigabit Ethernet interface 1/1 is configured as the transit link. The Gigabit Ethernet interface 1/1 allows the VLAN 46 to pass the transit traffic. It should also have non-default (other than 1) native VLAN (for example, VLAN 2), because the Cisco ENCS internal switch uplink (internal) has the native VLAN 1 configured.

Enable MSTP on all switches before isolating traffic.

```
switch
 interface gigabitEthernet1/0
  negotiation auto
  no shutdown
  switchport access vlan 1
  switchport mode access
  switchport trunk native vlan 1
  switchport trunk allowed vlan 1-2349,2450-4093
!
!
!
switch
 interface gigabitEthernet1/1
  negotiation auto
  no shutdown
spanning-tree mst 1 cost 200000000
 spanning-tree mst 2 cost 200000000
  switchport access vlan 46
  switchport mode trunk
  switchport trunk native vlan 2
  switchport trunk allowed vlan 1-2349,2450-4093
!
!
!
 spanning-tree enable
 spanning-tree mode mst
 spanning-tree mst configuration
 name     region1
  revision 1
  instance 1 vlan 1
  instance 2 vlan 46
 !
```

Use the **show switch vlan detailed** command to verify the configuration as shown below:

```
device# show switch vlan detailed

platform-detail hardware_info Manufacturer "Cisco Systems, Inc."
platform-detail hardware_info PID ENCS5412/K9
platform-detail hardware_info SN FGL212681GK
platform-detail hardware_info hardware-version M3
platform-detail hardware_info UUID 7BBEBDE0-CE3C-42E5-B564-CFEE8F18AE97
platform-detail hardware_info Version 3.8.1-FC3
platform-detail hardware_info Compile_Time "Sunday, April 15, 2018 [20:38:10 PDT]"
platform-detail hardware_info CPU_Information "Intel(R) Xeon(R) CPU D-1557 @ 1.50GHz 12
cores"
platform-detail hardware_info Memory_Information "16227148 kB"
platform-detail hardware_info Disk_Size "64.0 GB"
platform-detail hardware_info CIMC_IP NA
platform-detail hardware_info Entity-Name ENCS
platform-detail hardware_info Entity-Desc "Enterprise Network Compute System"
```

```
platform-detail software_packages Kernel_Version 3.10.0-514.21.1.1.el7.x86_64
platform-detail software_packages QEMU_Version 1.5.3
platform-detail software_packages LibVirt_Version 3.2.0
platform-detail software_packages OVS_Version 2.5.2
platform-detail switch_detail UUID NA
platform-detail switch_detail Type NA
platform-detail switch_detail Name NA
platform-detail switch_detail Ports 8
                                                                                PCI
NAME   TYPE      MEDIA         LINK  SPEED  MTU   MAC                 DETAIL
-------------------------------------------------------------------------------
GE0-0  physical  Twisted Pair  up    1000   9216  70:db:98:c3:f3:64   02:00.0
GE0-1  physical  Twisted Pair  up    1000   9216  70:db:98:c3:f3:65   02:00.1
MGMT   physical  Twisted Pair  up    1000   1500  70:db:98:c3:f3:d8   0e:00.0
```

Packet Flow for vBranch HA

This section explains high-level packet flow in failure and non-failure cases.

Non-Failure Case

In the non-failure case, both active and standby Cisco ENCS devices are up and running.

- LAN to WAN through the standby ENCS1 WAN link
  - The device in the LAN segment is configured with the default gateway as the HSRP virtual IP address, and since ENCS1 is an active HSRP, LAN traffic first comes to the active ENCS1.
  - LAN traffic goes through the service chain VM (Cisco ASAv), and then hits the Cisco ISRv. In this case, the destination IP is routable through the ENCS1 WAN interface. The Cisco ISRv sends traffic over the WAN link.
- LAN to WAN through the standby ENCS2 WAN link—In this case, the LAN to WAN traffic uses the transit link between the active and standby devices.
  - Devices in the LAN segment are configured with the default gateway as the HSRP virtual IP address, and since ENCS1 is an active HSRP, the LAN traffic first comes to the active ENCS1.
  - The LAN traffic goes through the service chain VMs (Cisco ASAv), and then hits the active Cisco ISRv. In this case, the destination IP is routable through the ENCS2 WAN interface. The traffic is sent to the Cisco ISRv on ENCS2 over the transit link, and then sent out over the WAN link to the destination.
- WAN to LAN through the active ENCS1
  - The WAN traffic hits the Cisco ISRv on ENCS1, then it goes through the service chain VMs, and sent to the LAN device.
- WAN to LAN through the standby ENCS2 WAN link—In this case, the WAN to LAN traffic uses the transit link between the active and standby devices.
  - The WAN traffic comes to the Cisco ISRv on ENCS2. The PBR/PFR configuration forces the traffic to use the transit link instead of the directly connected LAN port. So, the traffic is sent to the Cisco ISRv on ENCS1 over the transit link.

- Then, the traffic on ENCS1 goes through the service chain VMs, and sent to the LAN device.

Failure Case

In the failure case, the active device goes down, and the standby device becomes active.

The virtual IP (default gateway) address becomes active on ENCS2. The transit link will not be used. The traffic now goes through the service chain VMs on ENCS2, and gets forwarded directly between WAN and LAN interfaces. The PBR/PFR configuration should monitor the HSRP state, and use the LAN port instead of the transit link to forward LAN traffic.

Configuration Examples for vBranch HA

This sample configuration is for Cisco ENCS HA with a dual-WAN scenario. The Cisco ISRv is configured with vNICs connected to the wan-net, service-net, and transit link. HSRP is configured on the service-net interface. Each Cisco ENCS is provisioned with the Cisco ASAv (service-net) and Cisco vWAAS (service-net).



Note You can use this design as is, or modify as per the field requirement.

Example: Active Cisco ENCS Configuration with ISRv1

```
interface GigabitEthernet1
 vrf forwarding Mgmt-intf
 ip address 192.0.2.1 255.255.255.0
 negotiation auto
!
interface GigabitEthernet2
 description WAN-GE0-0-SRIOV-1
 ip address 192.0.2.2 255.255.255.0
 negotiation auto
!
interface GigabitEthernet3
 description LAN-NET
 no ip address
 shutdown
 negotiation auto
!
interface GigabitEthernet4
 description Service-NET-Virtio
 ip address 192.0.2.3 255.255.255.0
 standby use-bia
 standby 25 ip 192.0.2.20
 standby 25 timers 1 5
 standby 25 priority 105
 standby 25 preempt
 standby 25 track 1 decrement 10
 standby 25 track 2 decrement 10
 standby 25 track 3 decrement 10
 standby 25 track 5 shutdown
 standby 25 track 6 shutdown
 standby 25 track 7 shutdown
 negotiation auto
```

```
 bfd interval 9000 min_rx 9000 multiplier 3
!
interface GigabitEthernet5
 ip address 192.0.2.4 255.255.255.0
!
!
router eigrp stub 10
 network 25.25.25.0 0.0.0.255
 network 38.38.38.0 0.0.0.255
 network 46.46.46.0 0.0.0.255
  !
    !
track 1 ip sla 1 reachability
!
track 2 interface GigabitEthernet2 line-protocol
!
track 3 interface GigabitEthernet4 line-protocol
!
track 5 ip sla 5 reachability
!
track 6 ip sla 6 reachability
!
track 7 ip sla 7 reachability
!
ip sla 1
 icmp-echo 9.9.9.29 source-ip 192.0.2.2
 frequency 5
ip sla schedule 1 life forever start-time now
!
ip sla 5
 icmp-echo 25.25.25.11 source-ip 192.0.2.3
 frequency 5
ip sla schedule 5 life forever start-time now
!
ip sla 6
 icmp-echo 25.25.25.51 source-ip 192.0.2.3
 frequency 5
ip sla schedule 6 life forever start-time now
!
ip sla 7
 icmp-echo 25.25.25.75 source-ip 192.0.2.3
 frequency 5
ip sla schedule 7 life forever start-time now
!
event manager applet noshut_int
 event track 5 state up
 action 1.1 cli command "enable"
 action 1.2 cli command "config t"
 action 1.3 cli command "router eigrp 10"
 action 1.4 cli command "network 25.25.25.0 0.0.0.255"
 action 1.5 cli command "end"
event manager applet shut_int
 event track 5 state down
 action 1.1 cli command "enable"
 action 1.2 cli command "config t"
 action 1.3 cli command "router eigrp 10"
 action 1.4 cli command "no network 25.25.25.0 0.0.0.255"
 action 1.5 cli command "end"
event manager applet ASAv_noshut_int
 event track 6 state up
 action 1.1 cli command "enable"
 action 1.2 cli command "config t"
 action 1.3 cli command "router eigrp 10"
 action 1.4 cli command "network 25.25.25.0 0.0.0.255"
```

```
 action 1.5 cli command "end"
event manager applet ASAv_shut_int
event track 6 state down
 action 1.1 cli command "enable"
 action 1.2 cli command "config t"
 action 1.3 cli command "router eigrp 10"
 action 1.4 cli command "no network 25.25.25.0 0.0.0.255"
 action 1.5 cli command "end"
event manager applet vWAAS_noshut_int
 event track 7 state up
 action 1.1 cli command "enable"
 action 1.2 cli command "config t"
 action 1.3 cli command "router eigrp 10"
 action 1.4 cli command "network 25.25.25.0 0.0.0.255"
 action 1.5 cli command "end"
event manager applet vWAAS_shut_int
 event track 7 state down
 action 1.1 cli command "enable"
 action 1.2 cli command "config t"
 action 1.3 cli command "router eigrp 10"
 action 1.4 cli command "no network 25.25.25.0 0.0.0.255"
 action 1.5 cli command "end"
!
end
```

Example: Standby Cisco ENCS Configuration with ISRv2

```
interface GigabitEthernet1
 vrf forwarding Mgmt-intf
 ip address 192.0.2.1 255.255.255.0
 negotiation auto
!
interface GigabitEthernet2
 description WAN-GE0-0-SRIOV-1
 ip address 192.0.2.21 255.255.255.0
 negotiation auto
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
!
interface GigabitEthernet4
 description Service-NET-virtio
 ip address 192.0.2.22 255.255.255.0
 standby use-bia
 standby 25 ip 192.0.2.20
 standby 25 timers 1 5
 standby 25 preempt
 negotiation auto
 bfd interval 9000 min_rx 9000 multiplier 3
!
interface GigabitEthernet5
 ip address 192.0.2.23 255.255.255.0
!
!
router eigrp 10
network 8.8.8.0 0.0.0.255
 network 25.25.25.0 0.0.0.255
 network 46.46.46.0 0.0.0.255
```

!

Cisco ENCS Failure Points

Failure Points	Sequence of Events
ENCS chassis hardware failure: • Power down • Power cycle • Reboot Cisco Enterprise NFVIS software failure • Crash Cisco ISRv software failure • Stop (shutdown) • Reboot • Crash • Error	1. HSRP on ENCS2 detects the reachability failure to ENCS1, and triggers the failover. LAN virtual IP becomes active on ENCS2. 2. WAN-IP1 on ENCS1 becomes unreachable, and all the routes converge towards WAN-IP2 on ENCS2. WAN-IP2 is the only IP for branch connectivity. 3. All the WAN to LAN, and LAN to WAN traffic will now flow through ENCS2. 4. The PBR/PFR configuration will now select the LAN port as the preferred path instead of the transit link for the traffic heading to LAN.

ISRv1 (Active) Before the Failure

```
ISRv1# show platform software vnic-if interface-mapping
------------------------------------------------------------
 Interface Name        Driver Name         Mac Addr
------------------------------------------------------------
 GigabitEthernet5       i40evf             5254.003a.1020 (LAN-SRIOV-2)
 GigabitEthernet4       virtio             5254.0053.e392 (service-net)
 GigabitEthernet3       i40evf             5254.00c4.b925 (LAN-SRIOV-1)
 GigabitEthernet2       igbvf              5254.00d2.cc9a (GE0-0-SRIOV-1)
 GigabitEthernet1       virtio             5254.00d2.1b1c (int-mgmt-net)
------------------------------------------------------------

ISRv1# show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State   Active          Standby         Virtual IP
Gi4         25   105 P Active  local           192.0.2.1     192.0.2.222
#
```

ISRv2 (Standby) Before the Failure

```
ISRv2#show platform software vnic-if interface-mapping
------------------------------------------------------------
 Interface Name        Driver Name         Mac Addr
------------------------------------------------------------
 GigabitEthernet5       i40evf             5254.00cc.ce9f (LAN-SRIOV-2)
 GigabitEthernet4       virtio             5254.00e7.523f (Service-net)
```

```
 GigabitEthernet3       i40evf             5254.0055.ee45 (LAN-SRIOV-1)
 GigabitEthernet2       igbvf              5254.00a3.d443 (GE0-0-SRIOV-1)
 GigabitEthernet1       virtio             5254.0048.e84c (int-mgmt-net)
------------------------------------------------------------
```

```
ISRv2#show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State   Active          Standby         Virtual IP
Gi4         25   100 P Standby 192.0.2.20     local           192.0.2.222
```

ISRv2 After the Failure

ISRv1 becomes unreachable. ISRv2: The HSRP failover occurs, and the state changes from Standby to Active. The virtual IP (LAN side default gateway) becomes active on ENCS2 ISRv2.

```
ISRv2# show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State   Active          Standby         Virtual IP
Gi4         25   100 P Active  local           unknown         192.0.2.222
```

```
ISRv2# show logging
*Dec 13 21:22:17.138: %HSRP-5-STATECHANGE: GigabitEthernet4 Grp 25 state Speak -> Standby
*Dec 13 21:22:32.385: %HSRP-5-STATECHANGE: GigabitEthernet4 Grp 25 state Standby -> Active
```

Failure Points	Sequence of Events
WAN Net1 failure (WAN SRIOV VF connected to ISRv1) • Link down WAN Phy link failure • Switch failure • End-to-end connectivity failure	**1.** ISRv1 HSRP on ENCS1 detects the WAN connection failure. It reduces the LAN-HSRP priority. This failure is detected when the interface goes down due to VF going down or track object going down. **2.** WAN-IP1 becomes unreachable, and all the routes converge towards WAN-IP2 on ENCS2. WAN-IP2 is the only IP for branch connectivity. **3.** HSRP on ENCS2 becomes higher priority in the group, and takes over the active role. LAN-virtual IP becomes active on ENCS2. **4.** The PBR/PFR configuration will now select the LAN port as the preferred path instead of the transit link for the traffic destined to LAN. **5.** All the WAN to LAN, and LAN to WAN traffic will now flow through ENCS2.

ISRv1 After the Failure

ISRv1 becomes standby.

```
ISRv1# show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet1       192.0.2.1       YES NVRAM  up                    up
GigabitEthernet2       192.0.2.2     YES NVRAM  down                  down
```

```
GigabitEthernet3       unassigned      YES NVRAM  administratively down down
GigabitEthernet4       192.0.2.3     YES NVRAM  up                    up
GigabitEthernet5       unassigned      YES NVRAM  up                    up

ISRv1# show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State   Active          Standby         Virtual IP
Gi4         25   85  P Standby 192.0.2.22     local           192.0.2.222
ISRv1#
ISRv1#show logg
*Dec 14 03:41:52.307: %TRACK-6-STATE: 2 interface Gi2 line-protocol Up -> Down
*Dec 14 03:42:37.744: %HSRP-5-STATECHANGE: GigabitEthernet4 Grp 25 state Active -> Speak
*Dec 14 03:42:43.663: %HSRP-5-STATECHANGE: GigabitEthernet4 Grp 25 state Speak -> Standby
ISRv1#
ISRv1#show track
Track 1
  IP SLA 1 reachability
  Reachability is Down
    1405 changes, last change 00:03:08
  Latest operation return code: Timeout
  Tracked by:
    HSRP GigabitEthernet4 25
```

ISRv2 After the Failure

```
ISRv2# show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State   Active          Standby         Virtual IP
Gi4         25   100 P Active  local           192.0.2.3     192.0.2.222
```

Failure Points	Sequence of Events
LAN Phy link failure • Switch failure • End-to-end connectivity failure LAN connectivity failure • Switch failure • End-to-end connectivity failure SC Net failure (ISRv service-net down) • Link down VNFs (Cisco ASAv, Cisco vWAAS, and Windows/Linux) • Power down • Power cycle • Crash/reboot	**1.** ISRv1 HSRP on ENCS1 detects the LAN connection failure, and shut down the HSRP group. This failure is detected when the interface goes down due to the track object going down. **2.** EEM script on ISRv1 withdraws the routes (for example, delete EIGRP networks). All the branch traffic routes will now converge towards WAN-IP2 on ENCS2. WAN-IP2 is the only IP for branch connectivity. **3.** HSRP on ENCS-2 becomes active in the group. LAN virtual IP becomes active on ENCS2. **4.** On ISRv2, the PBR/PFR configuration will now select the LAN port as the preferred path, instead of the transit link for the traffic destined to LAN. **5.** All the WAN to LAN and LAN to WAN traffic will now flow through ENCS2.

ISRv1 After the Failure

```
ISRv1# show track
Track 7
  IP SLA 7 reachability
  Reachability is Down
    7 changes, last change 00:01:40
  Latest operation return code: Timeout
  Tracked by:
    HSRP GigabitEthernet3 25
    EEM 2450904616
    EEM 2450905656
ISRv1#

ISRv1# show ip eigrp topo
EIGRP-IPv4 Topology Table for AS(10)/ID(53.53.53.51)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

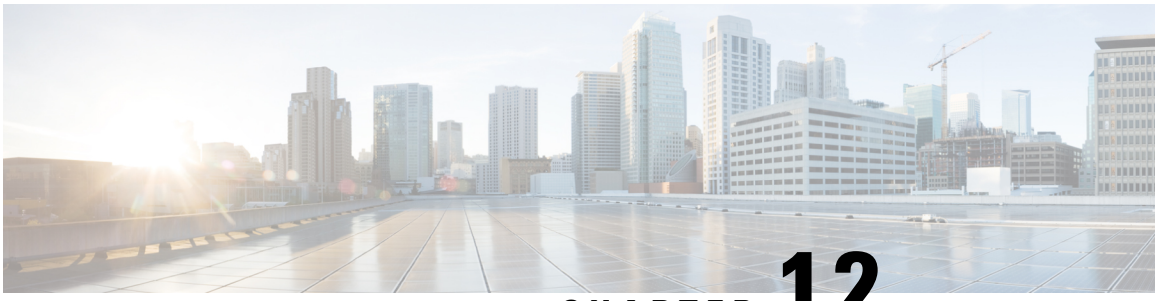
P 19.19.19.0/24, 1 successors, FD is 3328
        via 38.38.38.38 (3328/3072), GigabitEthernet2
P 9.9.9.0/24, 1 successors, FD is 3328
        via 38.38.38.38 (3328/3072), GigabitEthernet2
P 25.25.25.0/24, 0 successors, FD is Infinity
        via 38.38.38.38 (3840/3584), GigabitEthernet2
P 27.27.27.0/24, 1 successors, FD is 3328
        via 38.38.38.38 (3328/3072), GigabitEthernet2
P 38.38.38.0/24, 1 successors, FD is 2816
        via Connected, GigabitEthernet2
P 29.29.29.0/24, 1 successors, FD is 3072
        via 38.38.38.38 (3072/2816), GigabitEthernet2
P 33.33.33.0/24, 1 successors, FD is 3840
        via 38.38.38.38 (3840/3584), GigabitEthernet2
P 8.8.8.0/24, 1 successors, FD is 3584
        via 38.38.38.38 (3584/3328), GigabitEthernet2
P 53.53.53.0/24, 1 successors, FD is 2816
        via Connected, GigabitEthernet4
```

ISRv2 After the Failure

```
ISRv2# show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State   Active          Standby         Virtual IP
Gi3         25   100 P Active  local           unknown         192.0.2.222
```

Failure Points	Sequence of Events
WAN Net2 failure (WAN SRIOV VF connected to ISRv2 is down) • Link down	1. ISRv2 on ENCS2 detects the WAN connection failure. This failure is detected when the interface goes down due to VF going down or the track object going down. 2. WAN-IP2 becomes unreachable, and all the routes converge towards WAN-IP1 on ENCS1. WAN-IP1 is the only IP for branch connectivity. 3. All the WAN to LAN and LAN to WAN traffic will now flow through ENCS1.

Failure Points	Sequence of Events
Transit link between ENCS1 and ENCS2 fails • Link down	**1.** ISRv2 on ENCS2 detects the link going down due to VF going down or connection failure. The connection failure is detected by the track object. Then, ENCS2 WAN-IP2 link with EEM script is shut down. **2.** WAN-IP2 becomes unreachable, and all the routes converge towards WAN-IP1 on ENCS1. WAN-IP1 is the only IP for branch connectivity. **3.** All the WAN to LAN and LAN to WAN traffic will now flow through ENCS1.



CHAPTER 12

Cisco ENCS Single WAN IP Deployment Scenarios

- Single WAN IP Deployment, on page 127
- Preconfiguring the Cisco ENCS for a Single WAN IP Deployment, on page 128
- Single WAN IP Deployment with Gigabit Ethernet Interface 0/0, on page 129
- Single WAN IP Deployment with the 4G Interface, on page 130

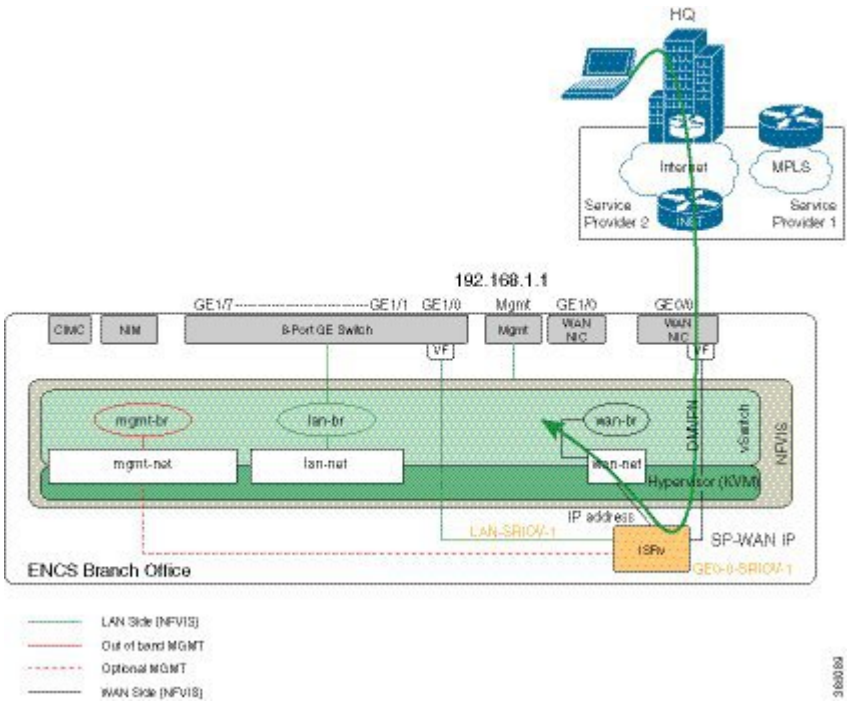
Single WAN IP Deployment

A single WAN IP deployment can be considered when the Cisco ENCS is preconfigured at the corporate main office with the service provider's WAN IP address, and shipped to the branch office for quick deployment. At the branch office, you do not have to perform any installation or configuration task. You just have to boot the system with the preconfigured setup. The single WAN IP deployment scenario could vary as per customer requirements. The following are two sample single WAN IP deployment scenarios with the Cisco ISRv:

Note Ensure that you preconfigure the Cisco ENCS at the main office before shippping the device to the branch office. You cannot connect to the remote branch office from your main office in a single WAN IP deployment scenario.

- Single WAN IP Deployment with Gigabit Ethernet Interface
- Single WAN IP Deployment with the 4G Interface

Figure 9: Single WAN IP Deployment Topology



Preconfiguring the Cisco ENCS for a Single WAN IP Deployment

To preconfigure the Cisco ENCS:

1. Install Cisco Enterprise NFVIS on the Cisco ENCS via CIMC. For details, see Installing Cisco Enterprise NFVIS on a Cisco ENCS 5100 and 5400, on page 12.
2. Connect your local system (laptop) to the local management interface of the host server.
3. Open the Cisco Enterprise NFVIS portal via https://192.168.1.1.
4. Upload the Cisco ISRv image using the portal, and register the VM.
5. From the portal, remove the default Gigabit Ethernet 0/0 or GE0-0 WAN interface.
6. Deploy Cisco ISRv with Gigabit Ethernet 2 for SRIOV-1 and Gigabit Ethernet 3 for the wan-net.
7. Open the Cisco ISRv VNC.
8. From the VNC console, configure ISRv Gigabit Ethernet 2 and Gigabit Ethernet 3 interfaces with appropriate IP addresses. Then, perform a "no shut" of the interfaces.
9. Set the WAN static IP address to be on the same subnet as ISRv Gigabit Ethernet 2 IP address, and use ISRv Gigabit Ethernet 2 interface IP address as the default gateway.
10. Ping with the Cisco ISRv IP address to ensure connectivity.
11. Configure Dynamic Multipoint VPN on the Cisco ISRv, and ensure the main server can access the portal.

For details, see the Dynamic Multipoint VPN Configuration Guide https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec_conn_dmvpn/configuration/xe-16/sec-conn-dmvpn-xe-16-book.html .

Single WAN IP Deployment with Gigabit Ethernet Interface 0/0

In this scenario, two Gigabit Ethernet interfaces are configured on the Cisco ISRv: Gigabit Ethernet2 as the outbound interface and Gigabit Ethernet3 as the internal interface. The outbound interface IP address is provided by the service provider. The internal interface is the WAN interface that serves as the default gateway for Cisco Enterprise NFVIS.

```
crypto isakmp policy 5
 authentication pre-share
 group 2
crypto isakmp key dmvpnkey address 0.0.0.0

crypto ipsec transform-set dmvpnset esp-3des esp-sha-hmac
 mode tunnel

crypto ipsec profile dmvpnprof
 set security-association lifetime seconds 1200
 set transform-set dmvpnset

! DMVPN tunnel configuration
interface Tunnel100
 ip address 192.0.2.3 255.255.255.0
 no ip redirects
 ip mtu 1440
 ip nhrp authentication dmvpnkey
 ip nhrp map 192.0.2.1 198.51.100.1
 ip nhrp network-id 90
 ip nhrp nhs 192.0.2.2
 tunnel source GigabitEthernet2
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile dmvpnprof
!
interface GigabitEthernet2
 description this is the outbound interface
 ip address 198.51.100.2 255.255.0.0

interface GigabitEthernet3
 description this is the inside interface
 ip address 192.0.2.10 255.255.255.0
!

router eigrp 90
 network 10.4.76.0 0.0.0.255
 network 192.0.2.1
 eigrp stub connected
 no auto-summary
!
ip route 20.1.0.0 255.255.0.0 198.51.100.1
!
Smart license configuration

ip name-server 198.51.100.9
ip domain lookup
service internal
do test license smart dev-cert Enable
```

```
service call-home
call-home
 contact-email-addr callhome@cisco.com
 mail-server 192.0.2.8 priority 1
 alert-group-config snapshot
  add-command "show license tech su"
 profile "CiscoTAC-1"
  active
  no destination transport-method email
  destination transport-method http
  no destination address http https://example.com/
  destination address http http://10.22.183.117:8080/ddce/services/DDCEService
!
clock timezone PST -7
ntp server 192.0.2.9
do license smart register idtoken NDM1NjE1MDAtNDViZC00ZTQ5LTg4MGEtNmRj
Njg2Mjg5ZDVlLTE0OTg5NDk2%0ANjEzNzd8elk5SEtoL2pMTGtuNSs3Q3Jxd
GVoSUVpTmFnY2l0alVqR3B5MzFj%0AVWVrST0%3D%0A
```

Single WAN IP Deployment with the 4G Interface

In this scenario, a 4G interface (NIM card) is configured as the outbound interface and Gigabit Ethernet3 as the internal interface. The outbound interface IP address is provided by the service provider. The internal interface is the WAN interface that serves as the default gateway for Cisco Enterprise NFVIS.

```
License Level: ax
License Type: N/A(Smart License Enabled)
Next reload license Level: ax

service timestamps debug datetime msec
service timestamps log datetime msec
service internal
service call-home
no platform punt-keepalive disable-kernel-core
platform console virtual
platform hardware throughput level MB 1000
!
hostname ISRv
!
boot-start-marker
boot system bootflash:isrv-universalk9.16.03.02.SPA.bin
boot-end-marker

clock timezone PST -7 0
call-home
 contact-email-addr callhome@cisco.com
 mail-server 192.0.2.8 priority 1
 alert-group-config snapshot
  add-command "show license tech su"
 profile "CiscoTAC-1"
  active
  destination transport-method http
  no destination transport-method email
  destination address http
http://198.51.100.4/Transportgateway/services/DeviceRequestHandler
  no destination address http
https://example.com/
!
ip name-server 198.51.100.2
```

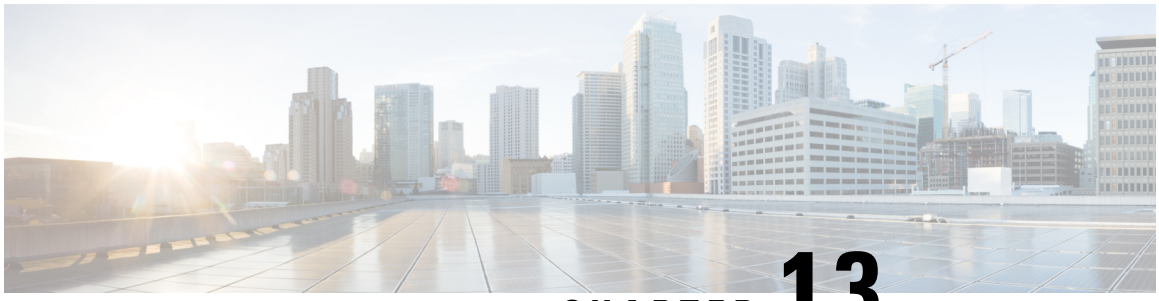
```
ip domain name cisco.com

! IPsec configuration

crypto isakmp policy 5
 authentication pre-share
 group 2
crypto isakmp key dmvpnkey address 0.0.0.0
!
!
crypto ipsec transform-set dmvpnset esp-3des esp-sha-hmac
 mode tunnel
!
!
crypto ipsec profile dmvpnprof
 set security-association lifetime seconds 1200
 set transform-set dmvpnset
!
!4G interface
controller Cellular 0/2/0
 lte modem link-recovery rssi onset-threshold -110
 lte modem link-recovery monitor-timer 20
 lte modem link-recovery wait-timer 10
 lte modem link-recovery debounce-count 6
!
!
!
no ip ftp passive
ip ftp username admin
ip ftp password admin
!DMVPN tunnel configuration

interface Tunnel100
 ip address 198.51.100.3 255.255.255.0
 no ip redirects
 ip mtu 1440
 ip nhrp authentication dmvpnkey
 ip nhrp map 198.51.100.5 192.0.2.7
 ip nhrp network-id 90
 ip nhrp nhs 198.51.100.5
 tunnel source Cellular0/2/0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile dmvpnprof
!
!
interface GigabitEthernet2
 ip address 198.51.100.6 255.255.255.0
 ip nat inside
 negotiation auto
!
interface GigabitEthernet3
 ip address 198.51.100.11 255.255.255.0
 negotiation auto
!
interface Cellular0/2/0
 ip address negotiated
 load-interval 30
 dialer in-band
 dialer idle-timeout 0
 dialer-group 1
 ipv6 address autoconfig
 pulse-time 1
!
```

```
interface Cellular0/2/1
 no ip address
!
!
router eigrp 90
 network 198.51.100.0 0.0.0.255
 network 198.52.100.0 0.0.0.255
 network 99.0.0.0
 eigrp stub connected
!
!
virtual-service csr_mgmt
 ip shared host-interface GigabitEthernet1
 activate
!
ip forward-protocol nd
ip http server
ip http authentication local
ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 Cellular0/2/0
ip route 192.0.2.12 255.255.255.0 198.51.100.5
ip route 192.0.2.13 255.255.255.255 198.51.100.5
ip route 192.0.2.14 255.255.255.255 198.51.100.5
ip route 192.0.2.15 255.255.255.255 198.51.100.5
ip route vrf Mgmt-intf 0.0.0.0 0.0.0.0 198.51.100.20
ip ssh authentication-retries 5
ip ssh rsa keypair-name ssh-key
ip ssh version 2
ip scp server enable
!
dialer-list 1 protocol ip permit
!
!
line con 0
 stopbits 1
line vty 0 4
 password cisco123
 login local
 transport input telnet ssh
!
ntp server 198.51.100.17
```



CHAPTER 13

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 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 reset must 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 until the system reboots successfully.

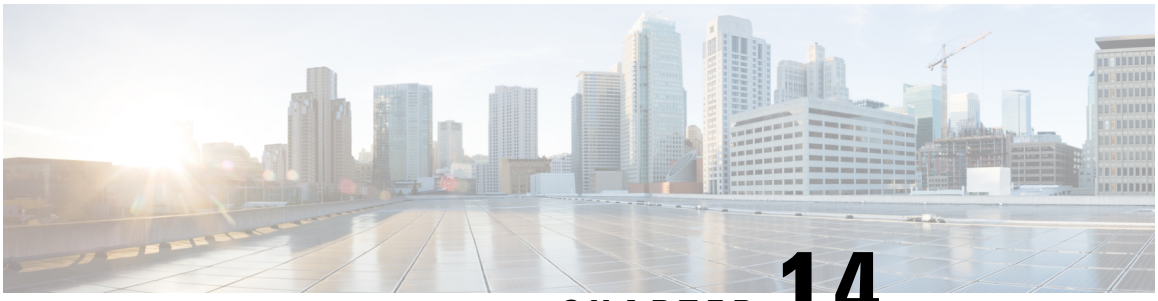
To reset to factory default:

```
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



CHAPTER 14

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.

- nfvisEvent, on page 136
- vmlcEvent, on page 153

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>
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>

Event Type	Notification Trigger	Notification Output Example
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>
FIPS_CONFIGURED_REBOOT_TO_ENABLE	When FIPS mode is configured but a reboot is required to enable it.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2020-06-07T21:26:06.231166+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>Fips-mode configured and a manual reboot is required to enable it</status_message> <details>NA</details> <event_type>FIPS_CONFIGURED_REBOOT_TO_ENABLE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
FIPS_ENABLED	FIPS mode is enabled.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2020-06-07T21:26:06.231166+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>Fips-mode successfully enabled</status_message> <details>NA</details> <event_type>FIPS_ENABLED</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
FIPS_FAILED		<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2020-06-07T21:26:06.231166+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>Fips-mode configuration is unsuccessful</status_message> <details>NA</details> <event_type>FIPS_FAILED</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
FIPS_DISABLED	FIPS mode is disabled.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2020-06-07T21:26:06.231166+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>Fips-mode successfully disabled</status_message> <details>NA</details> <event_type>FIPS_DISABLED</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
FIPS_UNCONFIGURED_REBOOT_TO_DISABLE		<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2020-06-07T21:26:06.231166+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>Fips-mode unconfigured and a manual reboot is required to disable it</status_message> <details>NA</details> <event_type>FIPS_UNCONFIGURED_REBOOT_TO_DISABLE</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.	

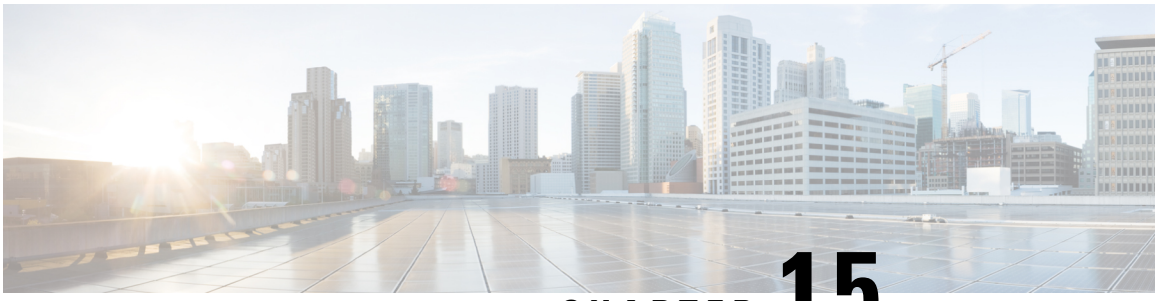
Event Type	Notification Trigger	Notification Output Example
		<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-12-08T00:50:39.034+00:00</eventTime> <vmlcEvent xmlns='http://www.cisco.com/nfvis/vm_lifecycle'> <status>SUCCESS</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>



CHAPTER 15

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 173

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 8: 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

• Syslog Messages, on page 173

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

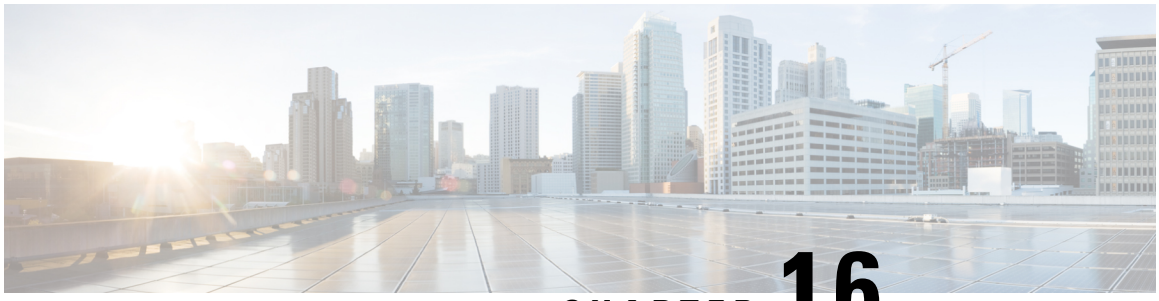
Event	Trigger Condition	Syslog Messages
MGMT_DHCP_RENEW	dhcp renew from MGMT interface	nfvis %SYS-6-MGMT_DHCP_RENEW: wan-br DHCP IP address is being renewed
INTF_STATUS_CHANGE	interface status change	nfvis %SYS-6-INTF_STATUS_CHANGE: Interface eth0, changed state to up
UPGRADE_REGISTER	upgrade package 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
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

Event	Trigger Condition	Syslog Messages
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
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

Event	Trigger Condition	Syslog Messages
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!
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

Event	Trigger Condition	Syslog Messages
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
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

Event	Trigger Condition	Syslog Messages
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
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 16

SNMP Support on NFVIS

- Introduction about SNMP, on page 179
- SNMP Operations, on page 179
- SNMP Versions, on page 181
- SNMP MIB Support, on page 182
- Configuring SNMP Support, on page 186

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

Table 9: Feature History

Feature Name	Release Information	Description
SNMP CISCO-MIB	NFVIS Release 4.11.1	The CISCO-MIB displays the Cisco NFVIS hostname using SNMP.
SNMP VM Monitoring MIB	NFVIS Release 4.4.1	Support added for SNMP VM monitoring MIBs.

The following MIBs are supported for SNMP on NFVIS:

CISCO-MIB starting from Cisco NFVIS Release 4.11.1:

CISCO-MIB OID 1.3.6.1.4.1.9.2.1.3. hostname

IF-MIB (1.3.6.1.2.1.31):

- 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 (1.3.6.1.2.1.47):

- entPhysicalIndex

- entPhysicalDescr
- entPhysicalVendorType
- entPhysicalContainedIn
- entPhysicalClass
- entPhysicalParentRelPos
- entPhysicalName
- entPhysicalHardwareRev
- entPhysicalFirmwareRev
- entPhysicalSoftwareRev
- entPhysicalSerialNum
- entPhysicalMfgName
- entPhysicalModelName
- entPhysicalAlias
- entPhysicalAssetID
- entPhysicalIsFRU

Cisco Process MIB (1.3.6.1.4.1.9.9.109):

- cpmCPUTotalPhysicalIndex (.2)
- cpmCPUTotal5secRev (.6.x)*
- cpmCPUTotal1minRev (.7.x)*
- cpmCPUTotal5minRev (.8.x)*
- cpmCPUMonInterval (.9)
- cpmCPUMemoryUsed (.12)
- cpmCPUMemoryFree (.13)
- cpmCPUMemoryKernelReserved (.14)
- cpmCPUMemoryHCUsed (.17)
- cpmCPUMemoryHCFree (.19)
- cpmCPUMemoryHCKernelReserved (.21)
- cpmCPULoadAvg1min (.24)
- cpmCPULoadAvg5min (.25)
- cpmCPULoadAvg15min (.26)

Note * indicates the support data required for a single CPU core starting from NFVIS 3.12.3 release.

Cisco Environmental MIB (1.3.6.1.4.1.9.9.13):

- Voltage Sensor:
  - ciscoEnvMonVoltageStatusDescr
  - ciscoEnvMonVoltageStatusValue
- Temperature Sensor:
  - ciscoEnvMonTemperatureStatusDescr
  - ciscoEnvMonTemperatureStatusValue
- Fan Sensor
  - ciscoEnvMonFanStatusDescr
  - ciscoEnvMonFanState

Note Sensor support for the following hardware platforms:

- ENCS 5400 series: all
- ENCS 5100 series: none
- UCS-E: voltage, temperature
- UCS-C: all
- CSP: CSP-2100, CSP-5228, CSP-5436 and CSP5444 (Beta)

Cisco Environmental Monitor MIB notification starting from NFVIS 3.12.3 release:

- ciscoEnvMonEnableShutdownNotification
- ciscoEnvMonEnableVoltageNotification
- ciscoEnvMonEnableTemperatureNotification
- ciscoEnvMonEnableFanNotification
- ciscoEnvMonEnableRedundantSupplyNotification
- ciscoEnvMonEnableStatChangeNotif

VM-MIB (1.3.6.1.2.1.236) starting from NFVIS 4.4 release:

- vmHypervisor:
  - vmHvSoftware

    - vmHvVersion
    - vmHvUpTime
- vmTable:
    - vmName
    - vmUUID
    - vmOperState
    - vmOSType
    - vmCurCpuNumber
    - vmMemUnit
    - vmCurMem
    - vmCpuTime
- vmCpuTable:
    - vmCpuCoreTime
- vmCpuAffinityTable
    - vmCpuAffinity

Configuring SNMP Support

Feature	Description
SNMP encryption passphrase	Starting from Cisco NFVIS Release 4.10.1, there is an option to add an optional passphrase for SNMP that can generate a different priv-key other than the auth-key.

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.
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 v3 user:

When security level is authPriv

```
configure terminal
snmp user <user_name> user-version 3 user-group <group_name> auth-protocol <auth>
priv-protocol <priv> passphrase <passphrase_string>
```

```
configure terminal
snmp user <user_name> user-version 3 user-group <group_name> auth-protocol <auth>
priv-protocol <priv> passphrase <passphrase_string> encryption-passphrase
<encryption_passphrase>
```

When security level is authNoPriv:

```
configure terminal
snmp user <user_name> user-version 3 user-group <group_name> auth-protocol <auth> passphrase
 <passphrase_string>
```

When security level is noAuthNopriv

```
configure terminal
snmp user <user_name> user-version 3 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.

Variables	Description
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\-_#@%$*&!].
encryption_passphrase	Passphrase string. Supporting string is [A-Za-z0-9\-_#@%$*&!]. The user must configure passphrase first to configure encryption-passphrase.



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 trap host:

```
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.

SNMP Configuration Examples

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 privprotocol aes
passphrase changePassphrase encryption-passphrase encryptPassphrase
! 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
!!
```

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 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

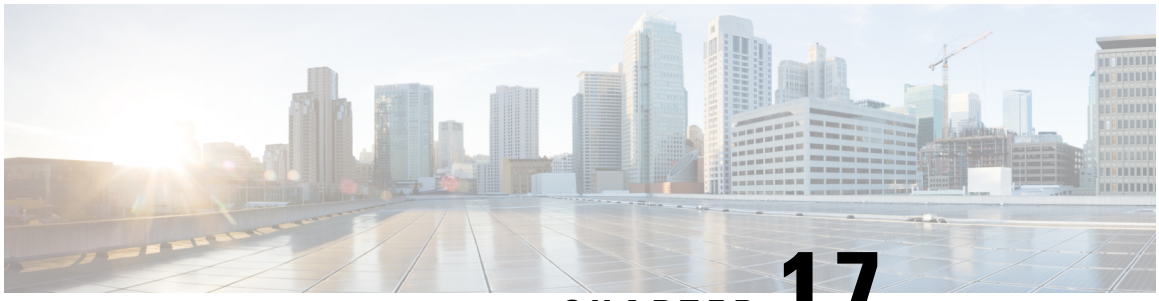
Upper limit for SNMP configurations:

- 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



CHAPTER 17

TACACS and RADIUS Support on NFVIS

- About RADIUS, on page 193
- RADIUS Operation, on page 193
- Configuring a TACACS+ Server, on page 194
- Configuring RADIUS, on page 195
- Default Authentication Order, on page 196

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.

Note You can configure upto four RADIUS servers. When multiple RADIUS servers are configured, if the first server is unreachable, NFVIS tries the next server in the order it is configured.

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 a TACACS+ Server

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.

For more details on TACACS+ configuration, see the Configuring TACACS module in TACACS+ Configuration Guide, Cisco IOS XE Release 3S.

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.

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

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

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 the same username is registered for both local authentication and authentication through RADIUS or TACACS+, RADIUS or TACACS+ is chosen as the authentication method.
- It is recommended to configure Syslog, on page 171 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

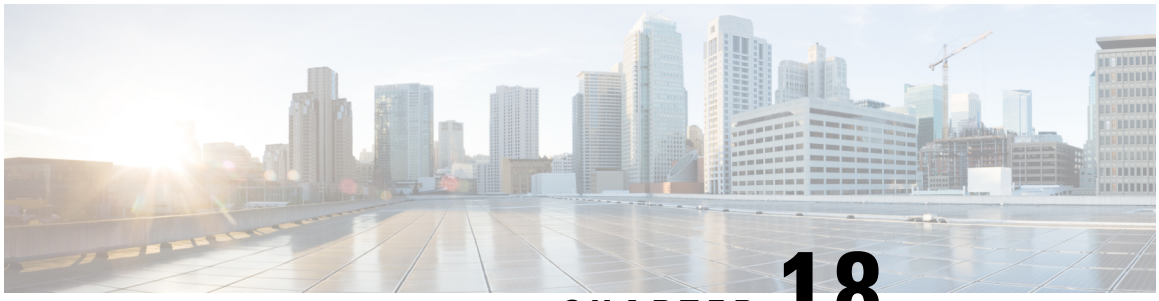
Starting from NFVIS 3.12.3 release, the supported aaa authentication order is local authentication followed by TACACS+.

- If the user is part of the local database, local authentication is executed and the user is permitted or denied access.
- If the user is not part of the local database, TACACS+ is used for authentication.
- If the same user is part of both the databases (local and TACACS+), the user can login with either the local password or the TACACS+ password. However, registering the same user in both the databases is not recommended.

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)#
```



CHAPTER 18

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.
- To configure any parameter on the ENCS switch portal, ensure that you add or remove parameters using separate transactions on the CLI. For example, adding and removing multiple VLANs from the ENCS switch portal is not supported in the same CLI transaction. Add a VLAN first and then remove the VLAN in a separate transaction.

- ENCS Switch Commands, on page 199
- ENCS Switch APIs, on page 199
- ENCS Switch Portal Configuration, on page 200

ENCS Switch Commands

See, Cisco Enterprise Network Compute System Switch Command Reference for switch commands.

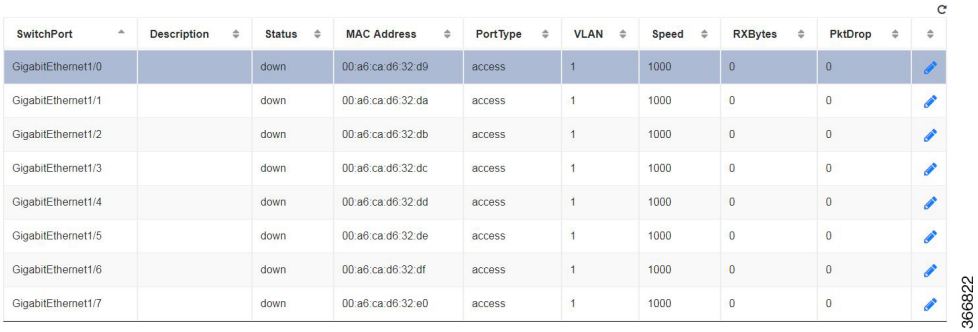
ENCS Switch APIs

See, API Reference for Cisco Enterprise Network Function Virtualization Infrastructure Software for switch related APIs.

ENCS Switch Portal Configuration

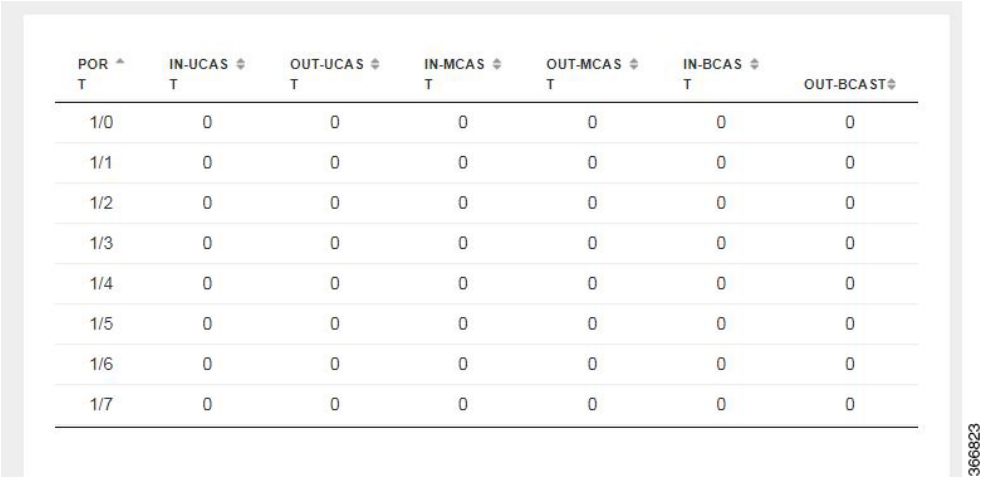
Switch Settings

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 10: Switch Settings Interface

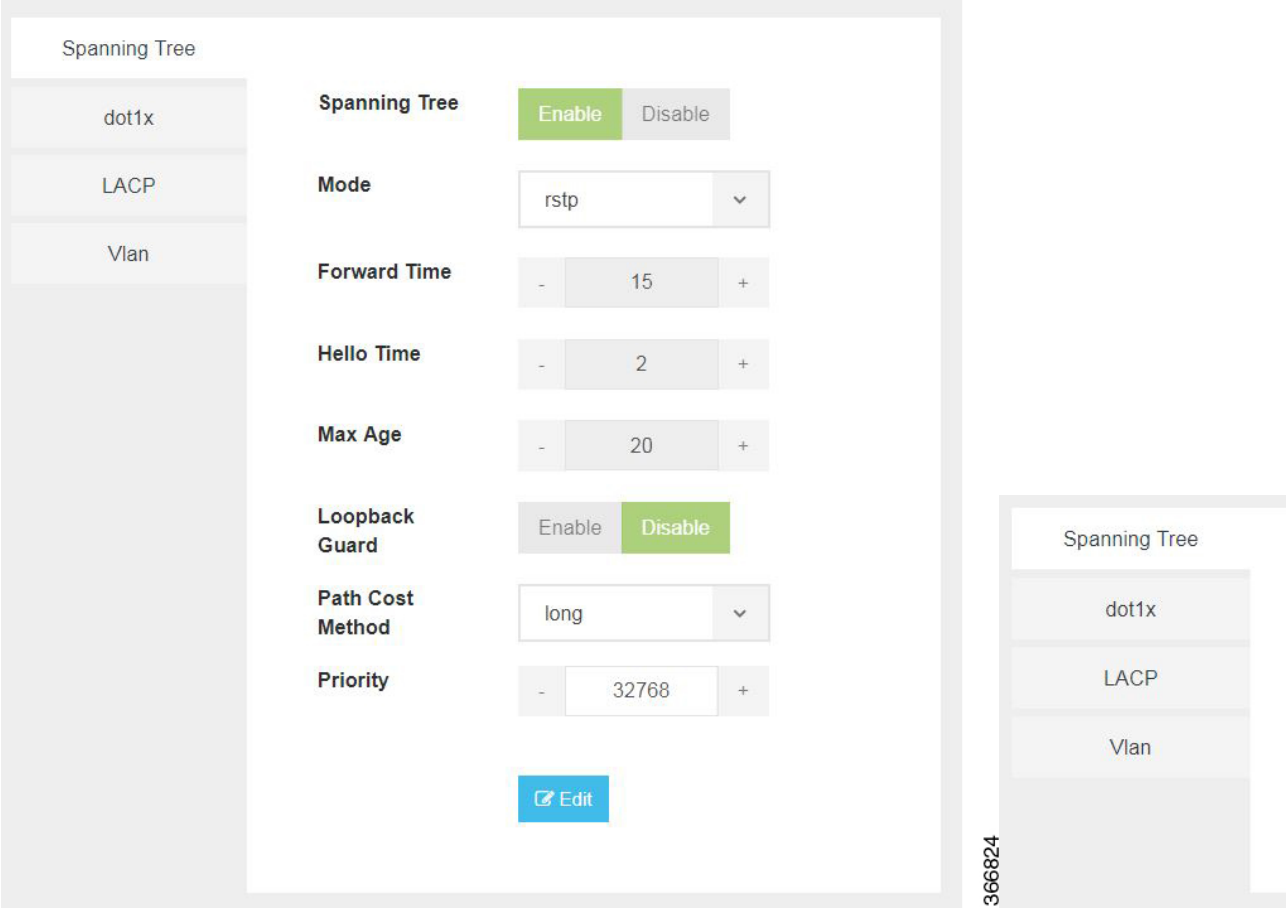
Parameter	Description	Values
SwitchPort	Specifies the switch interface name.	
Description	Specifies the description of the interface.	
Status	Specifies the status of the interface.	up or down
MAC Address	Specifies the MAC address of the interface.	

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:

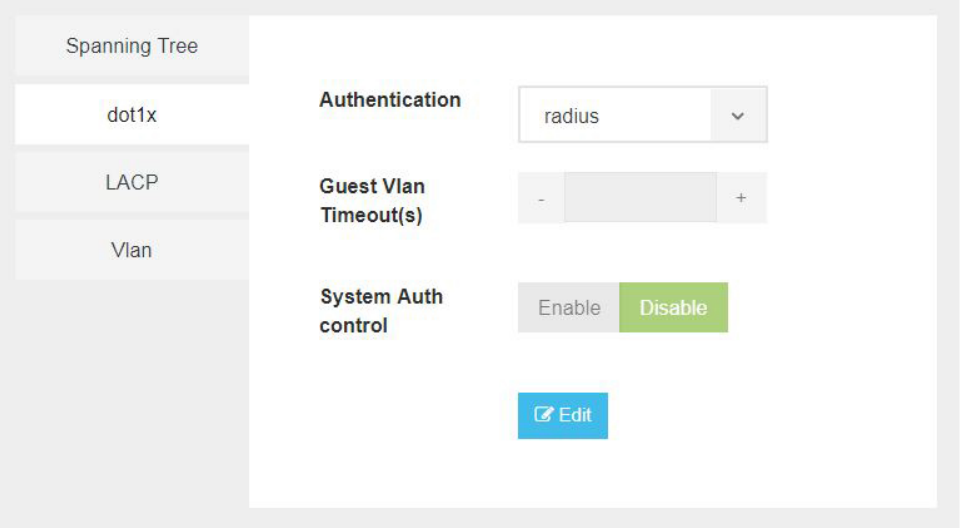
Table 11: Spanning Tree Parameters

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:

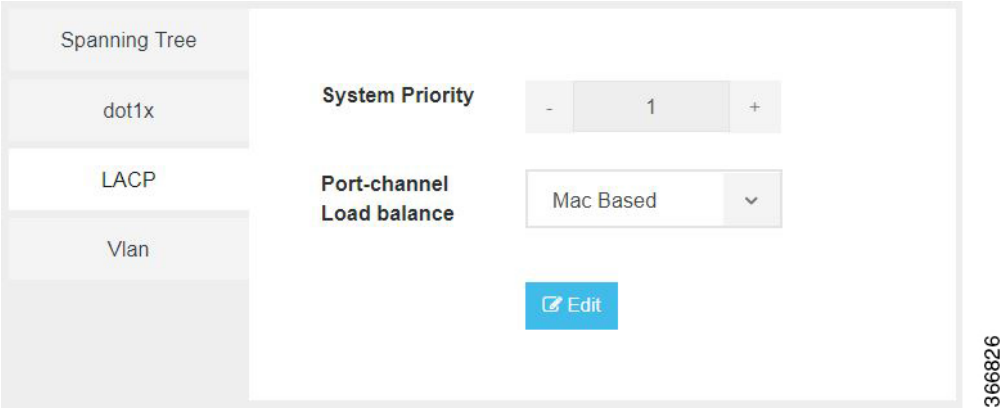
Table 12: Dot1x Parameters

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:

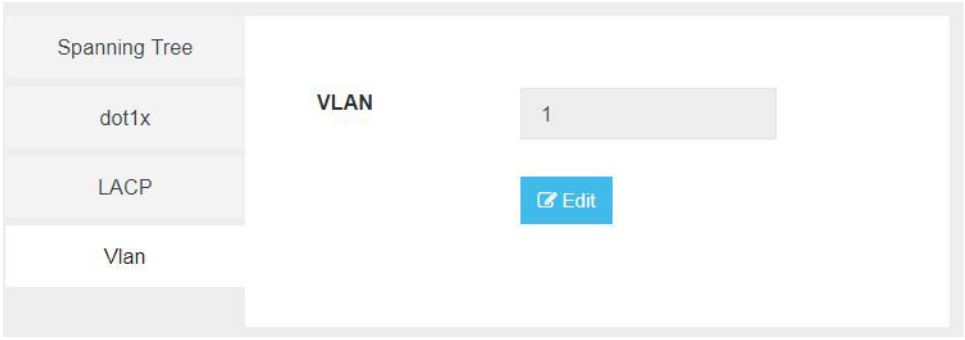
Table 13: LACP Parameters

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.

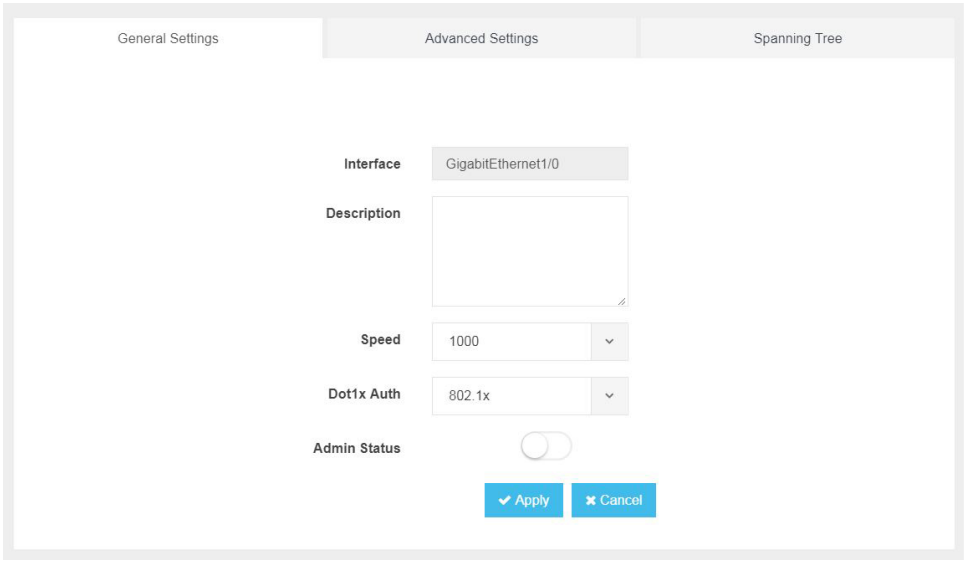
You can configure VLANs in the range **<1-2349>|<2450-4093>** for a specified switch port.



Note

- When you use multiple Virtual Network Inteface Cards (VNICs) in your VLAN, ensure that each of the VNICs are configured within the VLAN network.
- The Cisco ENCS hardware switch is typically connected to the Open vSwitch (OVS) using physical connections (like Ethernet cables) which are then represented as ports within the Open vSwitch configuration. When a VLAN is set up on the Cisco ENCS switch, perform the corresponding configuration on the OVS as well. Assign a VLAN tag to the OVS port that corresponds to the hardware switch port.
- Use native VLANs for untagged traffic. If a hardware switch port is configured to use a certain VLAN as native, corresponding OVS port should also be configured to tag the untagged incoming traffic with the same VLAN. Allowed VLAN lists define which VLANs are allowed on a port.
- While a hardware switch and OVS are logically connected, configurations do not automatically propagate between the two. Manually configure the both to ensure consistency and correct networking behavior.

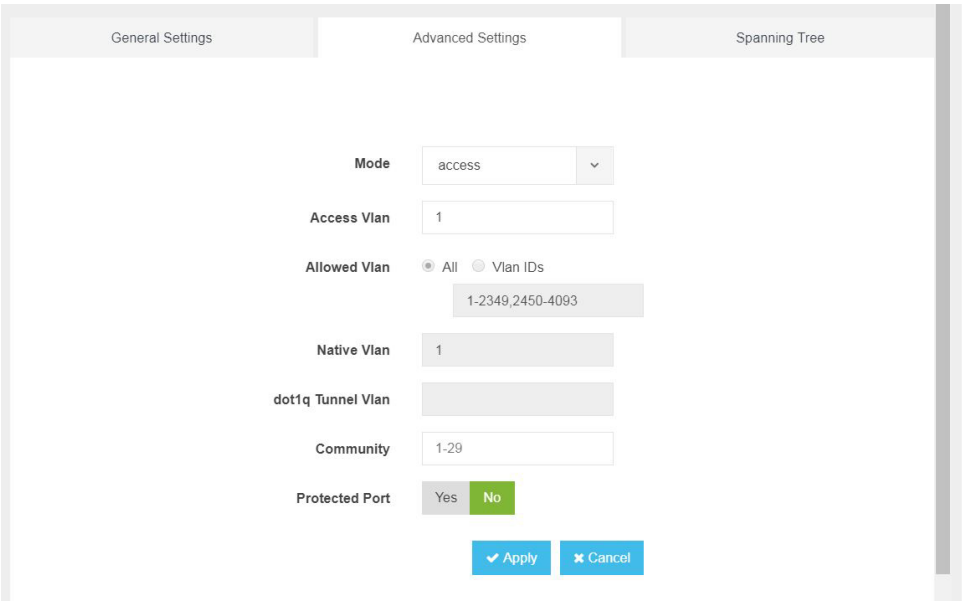
Configuring General Settings



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

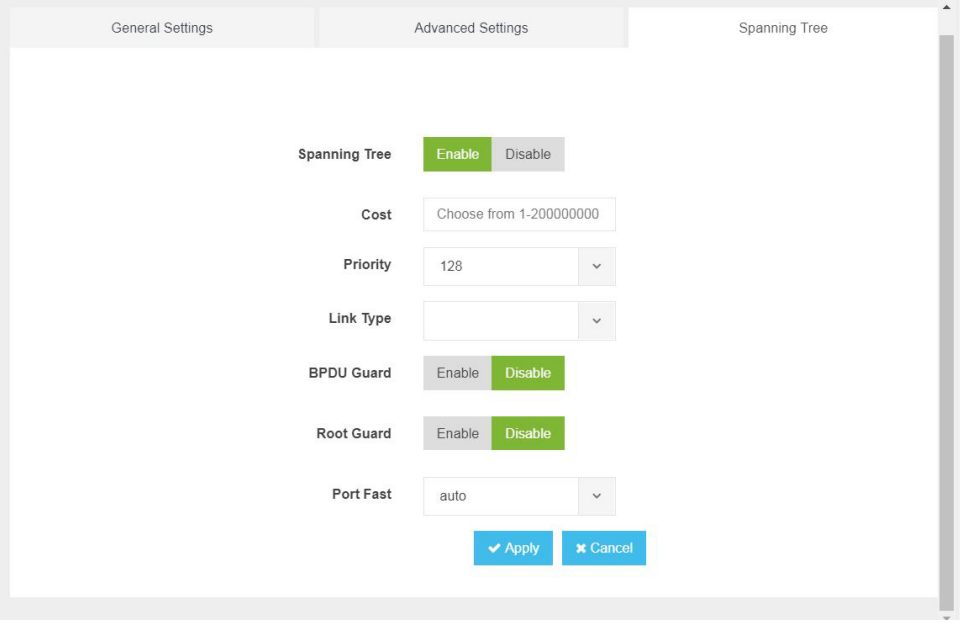


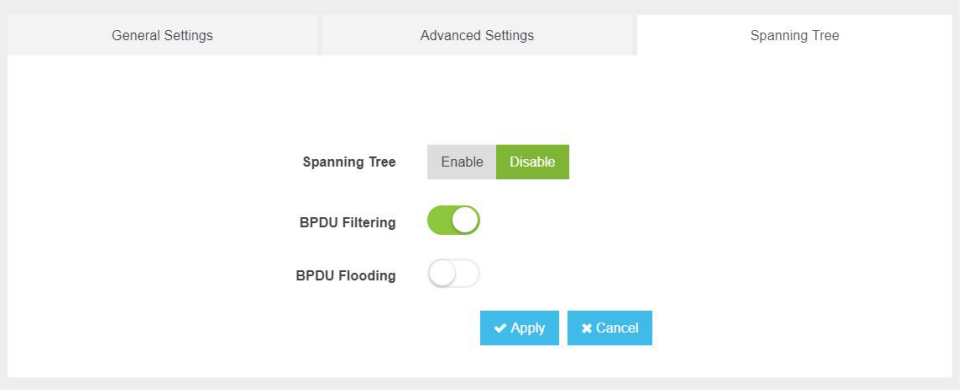
You can make the advanced settings using the following parameters for each switch interface:

- 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 204.

Configuring Spanning Tree per Interface

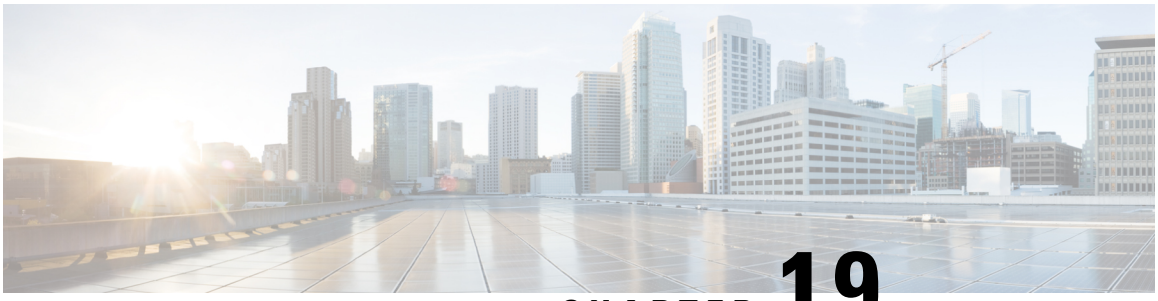




You can configure spanning tree for each switch interface using the following parameters:

- 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



CHAPTER 19

Configuring Secondary IP Address and Source Interface

Secondary IP Address

The Cisco Enterprise NFVIS supports multiple IP addresses per interface. You can configure a secondary IP address on the WAN interface, as an additional IP address to reach the software. Set the external routes for secondary IP address 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 address through ISRv, the WAN physical port is removed from wan-br similar to single IP address.

To configure Secondary IP Address:

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

Source Interface

The Source Interface feature lets you assign an IP address to a source interface.. The IP address configured is used for packets generated by the NFVIS. The packets generated use the default route.

Prerequisites for configuring Source Interface

- The IP address must be one of the IP addresses configured 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 address is accepted but cannot be applied. The configuration takes effect once you switch from DHCP to static.

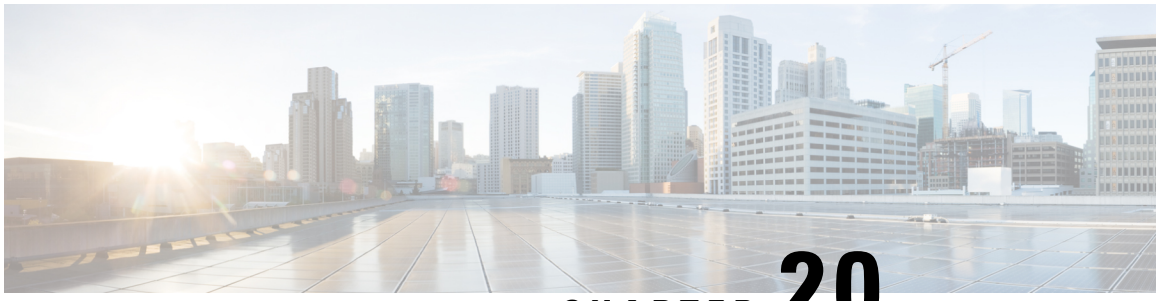
To configure source Interface:

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

The secondary IP address and source interface related errors are logged in **show log nfvis_config.log** file.

Secondary IP Address 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



CHAPTER 20

Ports and Port Channels

This chapter contains the following sections.

- Configuring Port Channels, on page 211
- Configuring LLDP, on page 217
- Configuring Administrative Status of a Port, on page 218
- Tracking Changes for a Port, on page 218
- Speed, Duplex and Autonegotiation, on page 219

Configuring 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.



Note

- The Physical Network Interface Controllers (PNICs) added to the port channel should be uniform. For example, all the PNICs associated with the port channel must have SRIOV VFs or they should not have SRIOV VFs.
- The Data Plane Development Kit (DPDK) can be associated only with port channels that have no SRIOV VFs attached to them. When a port channel is attached to a bridge and if the port channel has SRIOV VFs attached, the bridge gets automatically downgraded to a non-DPDK bridge.

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.

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
```



Note Ensure to commit the changes.

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
```



Note Ensure to commit the changes.

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



Note Ensure to commit the changes.

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
```



Note Ensure to commit the changes.

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
```



Note Ensure to commit the changes.

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



Note Ensure to commit the changes.

2. Remove port channel from the bridge.

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



Note Ensure to commit the changes.

3. Delete port channel.

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



Note Ensure to commit the changes.

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

Adding a Port to a Port Channel

A port channel must have at least two ports. A bond is created when a port channel has more than two members and the port channel is added to a bridge. 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:

```
configure terminal
pnic eth1 member_of pc
commit
```

Adding a Port to a Port Channel APIs and Commands

APIs	Commands
/api/config/pnics/pnic/*name*/member_of	pnic *name* member_of *portchannel_name*

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 pc
commit
```

Adding a Port Channel to a Bridge APIs and Commands

APIs	Commands
/api/config/bridges/bridge/*bridgename*	bridges bridge *name* port *portchannel_name*

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.

To delete a port channel:

```
configure terminal
no pnic pc
commit
```

Port Channel Deletion APIs and Commands

APIs	Commands
/api/config/pnics/pnic/*portchannel_name*	no pnic *portchannel_name*
/api/operational/pnics	show pnic

Removing a Port from a Port Channel

To remove a port from a port channel:

```
configure terminal
no pnic eth1 member_of pc
commit
```

Removing a Port from a Port Channel APIs and Commands

APIs	Commands
/api/config/pnics/pnic/*name*/member_of	no pnic *name* member_of *portchannel_name*

Removing a Port Channel from a Bridge

To remove a port channel from a bridge:

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

Removing a Port Channel from a Bridge APIs and Commands

APIs	Commands
/api/config/bridges/bridge/*bridgename*	no bridges bridge *bridgename* port *portname*

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 their interfaces 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

Administrative 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.

Note Administrative status cannot be enabled on port channel.

To configure the admin status 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

Note Speed and duplex values in **show pnic** and **ethtool** outputs may differ depending on the peer device's interface speed and duplex settings.

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 10 G.

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 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
```

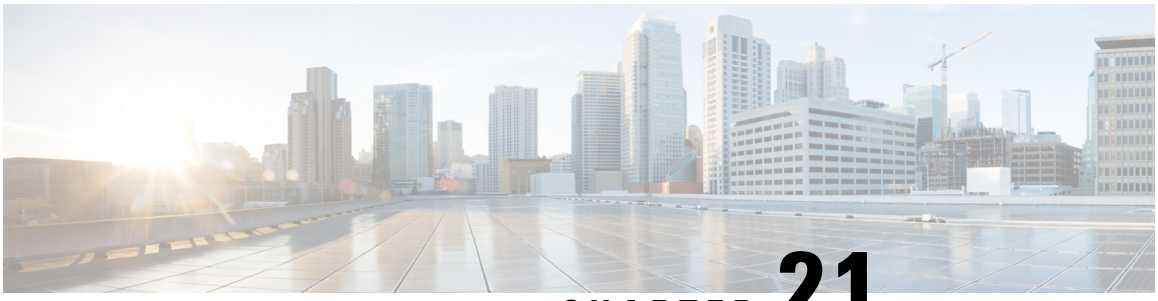
To verify the PNIC speed and duplex configurations, use the **show notification stream nfvis Event** command.

```
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



CHAPTER 21

MSTP for ENCS 5400 8-Port Switch

Multiple Spanning Trees Protocol (MSTP) is introduced to the 8-port switch on ENCS 5400. MSTP enables multiple VLANs to be mapped to the same spanning tree instance, which reduces the number of spanning-tree instances needed to support a large number of VLANs.

ENCS 5400 switch supports 15 instances. Each spanning tree instance is identified by an instance ID from 1 to 15.

To enable MST:

```
configure terminal
switch
    spanning-tree mode mst
    commit
```

To configure VLAN-to-instance mapping:

```
configure terminal
switch
    spanning-tree mst configuration
        instance 1 vlan 15
        name mst_test
        revision 2
        commit
```

To configure the switch priority:

```
configure terminal
switch
    spanning-tree mst 1 priority 0
    commit
```

To configure the path cost per interface's mst instance:

```
configure terminal
switch
    interface gigabitEthernet 1/1
    spanning-tree mst 1 port-priority 96
    spanning-tree mst 1 cost 2
    commit
```

To configure the maximum hop count for all MST instances:

```
configure terminal
```

```
switch
    spanning-tree mst max-hops 10
    commit
```

Use the **show switch spanning-tree mstp configuration global** command to verify the MSTP configuration.

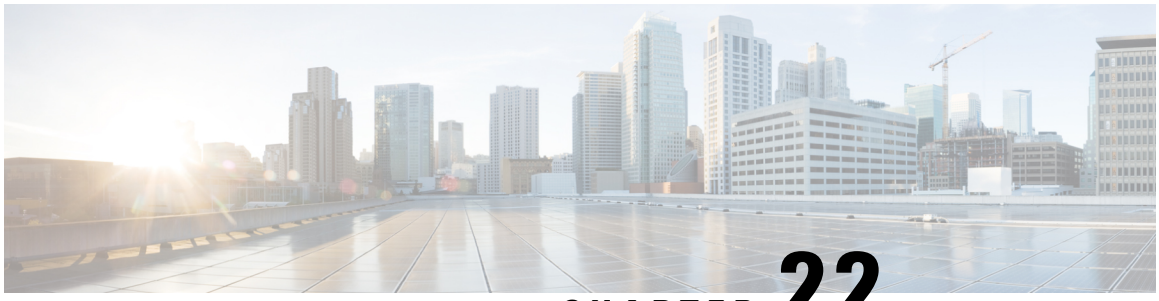
```
nfvis# show switch spanning-tree mstp configuration global
spanning-tree mstp configuration global name transit-net
spanning-tree mstp configuration global revision 2
spanning-tree mstp configuration global max-hops 20
```

Use the **show switch spanning-tree mstp configuration instance-list** command to verify the MSTP instance configuration.

```
nfvis# show switch spanning-tree mstp configuration instance-list
INSTANCE  VLANS MAPPED      STATE
------------------------------------
0         1,2350-2353,2363  enabled
1         15                enabled
```

To display the global information for MSTP instance 2 use **show switch spanning-tree mstp summary instance-global-info** command.

To display interface information for MSTP instance 2 use **show switch spanning-tree mstp summary instance-interface-info 2** command



CHAPTER 22

ENCS 5400 Switch LLDP

This chapter describes how to configure the Link Layer Discovery Protocol (LLDP) on ENCS 5400.

- Link Layer Discovery Protocol (LLDP) Commands, on page 225
- Enabling and Disabling LLDP, on page 225
- Configuring LLDP Characteristics, on page 226

Link Layer Discovery Protocol (LLDP) Commands

Link Layer Discovery Protocol (LLDP), is a neighbor discovery protocol that is used for network devices to advertise information about themselves to other devices on the network. This protocol runs over the data-link layer, which allows two systems running different network layer protocols to learn about each other.

LLDP is unidirectional, operating only in an advertising mode. LLDP does not solicit information or monitor state changes between LLDP nodes. LLDP periodically sends advertisements to a constrained multicast address. Devices supporting LLDP can send information about themselves while they receive and record information about their neighbors. Additionally, devices can choose to turn off the send or receive functions independently. Advertisements are sent out and received on every active and enabled interface, allowing any device in a network to learn about all devices to which it is connected.

LLDP supports a set of attributes that it uses to discover neighbor devices. These attributes contain type, length, and value descriptions and are referred to as TLVs. LLDP supported devices can use TLVs to receive and send information to their neighbors. Details such as configuration information, device capabilities, and device identity can be advertised using this protocol.

Enabling and Disabling LLDP

To globally enable LLDP:

```
configure terminal
switch
    lldp run
    commit
```

To globally disable LLDP:

```
configure terminal
switch
```

```
    no lldp run
    commit
```

LLDP is enabled by default on all supported interfaces. You must enable LLDP globally to allow a device to send LLDP packets. However, no changes are required at the interface level.

You can configure the interface selectively not to send and receive LLDP packets with the **no lldp transmit** and **no lldp receive** commands.

To enable LLDP on an interface:

```
configure terminal
switch
    interface gigabitEthernet1/0
        lldp transmit
        lldp receive
        commit
```

Configuring LLDP Characteristics

To specify an interval at which LLDP packets are sent:

```
configure terminal
switch
    lldp timer 135
    commit
```

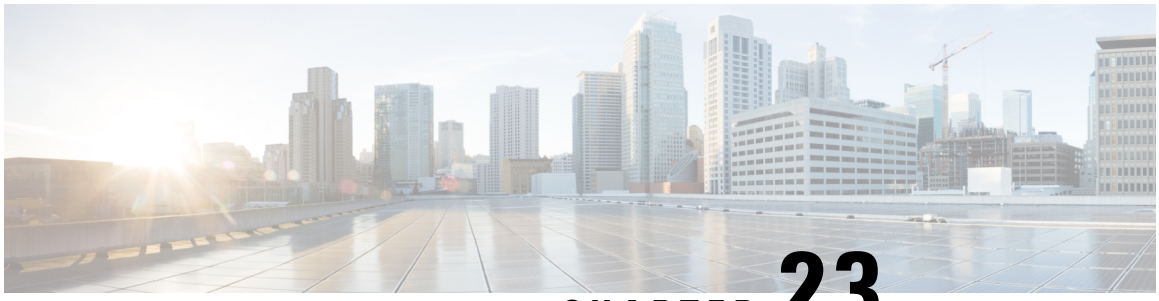
To display LLDP statistics:

```
nfvis# show switch lldp statistics
                                                                        RX
      TX             RX FRAMES          RX         TLVS           AGEOUTS
PORT  FRAMES  TOTAL  DISCARDED  ERRORS  DISCARDED  UNRECOGNIZED  TOTAL
-------------------------------------------------------------------------
1/0   0       0      0          0       0          0             0
1/1   0       0      0          0       0          0             0
1/2   792     756    0          0       0          0             0
1/3   791     756    0          0       0          0             0
1/4   0       0      0          0       0          0             0
1/5   0       0      0          0       0          0             0
1/6   792     756    0          0       0          0             0
1/7   791     756    0          0       0          0             0
```

To display information about LLDP neighbors:

```
nfvis# show switch lldp neighbors

SYSTEM
INDEX PORT DEVICE ID PORT ID NAME CAPABILITIES TTL
-----------------------------------------------------------------------
1 gi1/1 00:1a:6c:81:f0:80 Gi1/0/31 SW-026 Bridge 93
2 gi1/6 2c:0b:e9:3c:89:00 Gi1/0/5 Switch Bridge 119
```



CHAPTER 23

Secure Overlay and Single IP Configuration

- Secure Overlay, on page 227
- Single Public IP Address and Secure Overlay, on page 237
- Single IP Address Without Secure Overlay, on page 242

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 network. IPSec is a framework with protocols and algorithms to provide secured data transmission 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

- Introduced in NFVIS 4.2.1 release - aes256-sha512-modp2048
- Introduced in NFVIS 4.2.1 release - aes256-sha512-modp4096
- Introduced in NFVIS 4.4.1 release - aes128-sha1-modp1536, aes256-sha1-modp2048, and aes256-sha256-modp2048

- 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).
  - Introduced in NFVIS 4.4.1 release - local system IP address can be learnt through IPSec negotiation when local system IP address is not configured.

- Local system IP subnet:
  - local system IP subnet prefix.
  - Introduced in NFVIS 4.4.1 release - local system IP address and subnet can be learnt through IPSec negotiation. In this case local system IP subnet should not be configured.

- Local bridge for NFVIS reaching out to remote VPN server:
  - wan-br by default
  - wan2-br

- Local bridge and Dual local bridge:

  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

- Introduced in NFVIS 4.4.1 release - Active-Standby headend VPN responder scenario is supported. You can configure a list of Remote IDs, each of which is corresponding to one remote VPN responder.
- Introduced in NFVIS 4.5.1 release - Remote ID configuration using EAP authentication is simplified. If the added security using a distinguished name is not required when using EAP authentication, then an FQDN can be configured on NFVIS to simplify the remote ID configuration and reduce authentication complexity.

- Remote Interface IP address:
  - IP address
  - Introduced in NFVIS 4.4.1 release - FQDN
  - For the FQDN, DNS server has to be configured in system settings or learned through DHCP.

- Remote system IP address:
  - IP address
  - Introduced in NFVIS 4.4.1 release - Active-Standby headend VPN responder scenarios is supported. You can configure a list of Remote system IP addresses, each of which is corresponding to a tunnel IP address on a remote VPN responder.
  - Introduced in NFVIS 4.5.1 release - Remote system IP address can be learned from the remote VPN responder. To do so, leave the **remote-system-ip-addr** field out the of secure overlay configuration entirely

- Remote system IP Subnet:
  - User can specify IP subnet prefix.
  - Introduced in NFVIS 4.1.1 release - User can specify 0.0.0.0/0.
    - Default route is set to Secure Overlay tunnel
    - Following routes bypass the tunnel:
      - If PNP server IP address is discovered before Secure Overlay is configured
      - If DNS server IP address is set before Secure Overlay is configured.
      - Any NFVIS local static route
      - Any route added by user using **system routes route** command

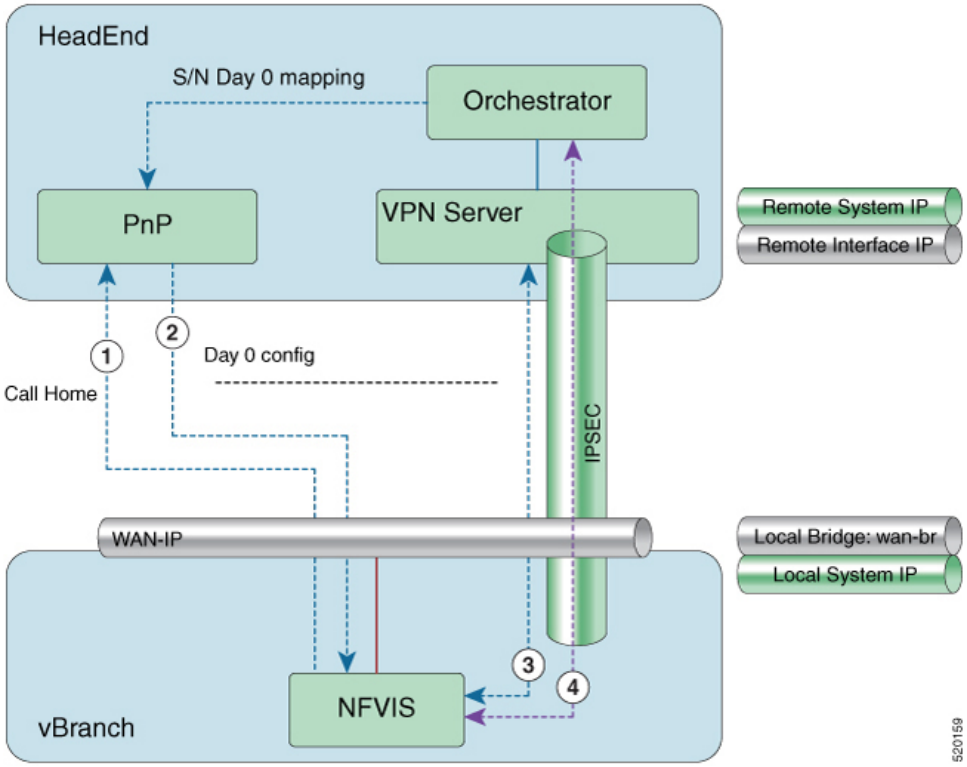
  - Introduced in NFVIS 4.4.1 release - Active-Standby headend VPN responder scenario is supported. You can configure a list of Remote system IP subnets, each of which corresponds to a tunnel IP address on a remote VPN responder.

- BGP Neighbor Name:
  - Introduced in NFVIS 4.4.1 release, Secure Overlay feature can work together with BGP feature. BGP session can be established over IPSec tunnel. NFVIS can learn private remote subnets from BGP neighbor over the tunnel. These learnt private subnets are added to the routing table for IPsec tunnel. BGP Neighbor Name allows BGP neighbor session to be established with the active secure overlay remote system IP address if the neighbor name is also configured under **router bgp**

configuration. NFVIS will automatically determine which remote system IP address belongs to the active remote IPSec VPN responder.

- Introduced in NFVIS 4.5.1 release - Secure overlay feature works together with BGP feature to announce NFVIS subnet routes over an IPSec tunnel to a BGP neighbor.

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 myconn
    local-system-ip-addr 27.27.27.1
    local-system-ip-bridge int-mgmt-net
    local-id mail@gmail.com
    remote-interface-ip-addr nfvisoverlay.cisco.com
    remote-system-ip-addr [ 166.34.121.111 166.34.121.112 ]
    psk local-psk Cisco1234Admin
```

```
    psk remote-psk Cisco1234Admin
     !

confirgure terminal
secure-overlay myconn
local-system-ip-addr 28.28.28.1
local-system-ip-subnet 28.28.28.0/24
local-system-ip-bridge int-mgmt-net
local-id AxxxY@cisco.com
remote-interface-ip-addr C*****d.cisco.com
remote-system-ip-addr [ 166.35.121.112 166.34.121.112 ]
remote-system-ip-subnet [ 166.35.121.112/32 166.34.121.112/32 ]
remote-id [
CN=vbranch,unstructuredAddress=10.30.1.114,unstructuredName=csr-vpn-srvr-02.cisco.com
CN=vbranch,unstructuredAddress=10.30.1.153,unstructuredName=csr-vpn-srvr-03.cisco.com ]
ike-cipher [ aes256-sha512-modp2048 ]
esp-cipher [ aes256-sha512-modp2048 ]
eap username admin
eap password  Cisco123#
eap cacert intdatastore:uploads/ca.pem
!
```

To get the secure overlay state:

```
nfvis# show secure-overlay
                                        ACTIVE      ACTIVE

               ACTIVE           SELECTED  LOCAL       REMOTE       ACTIVE REMOTE

               LOCAL   STATE    LOCAL     SYSTEM IP   INTERFACE    SYSTEM IP       ACTIVE
REMOTE

NAME    STATE  BRIDGE  DETAILS  BRIDGE    ADDR        IP ADDR      ADDR            SYSTEM
IP SUBNET   ACTIVE REMOTE ID

-------------------------------------------------------------------------------------------------------------------------------------

myconn  up     wan-br  -        wan-br    28.28.28.1  10.30.1.153  166.34.121.112
166.34.121.112/32
"CN=vbranch,unstructuredAddress=10.30.1.153,unstructuredName=csr-vpn-srvr-03.cisco.com"

nfvis#
```

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

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <secure-overlays xmlns="http://www.cisco.com/nfvis/secure-overlay">
    <secure-overlay>
      <name>myconn</name>
      <local-system-ip-addr>28.28.28.1</local-system-ip-addr>
      <local-system-ip-subnet>28.28.28.0/24</local-system-ip-subnet>
      <local-system-ip-bridge>int-mgmt-net</local-system-ip-bridge>
      <local-id>branch1@vpntest.com</local-id>
```

```
      <remote-interface-ip-addr>csrhead.cisco.com</remote-interface-ip-addr>
      <remote-system-ip-addr>166.35.121.112</remote-system-ip-addr>
      <remote-system-ip-addr>166.34.121.112</remote-system-ip-addr>
      <remote-system-ip-subnet>166.35.121.112/32</remote-system-ip-subnet>
      <remote-system-ip-subnet>166.34.121.112/32</remote-system-ip-subnet>

<remote-id>CN=vbranch,unstructuredAddress=10.30.1.114,unstructuredName=csr-vpn-srvr-02.cisco.com</remote-id>


<remote-id>CN=vbranch,unstructuredAddress=10.30.1.153,unstructuredName=csr-vpn-srvr-03.cisco.com</remote-id>

      <ike-cipher>aes256-sha512-modp2048</ike-cipher>
      <esp-cipher>aes256-sha512-modp2048</esp-cipher>
      <eap>
        <username>admin</username>
        <password>$7$ZEh54XQvvwwTicwX+lEuMWjuA7q+sfaa</password>
        <cacert>intdatastore:uploads/ca.pem</cacert>
      </eap>
    </secure-overlay>
  </secure-overlays>
</config>
```

VPN configuration on VPN server:

```
aaa new-model

!

aaa group server radius radius-group

 server-private 10.30.1.234 auth-port 1812 acct-port 1813 key Cisco123#

 ip vrf forwarding public-vrf

 ip radius source-interface GigabitEthernet2

!

aaa authentication login default group radius-group local

aaa authentication login ucpe-authen group radius-group

aaa authorization console

aaa authorization exec default local

aaa authorization network default local

!

aaa session-id common

!

crypto pki trustpoint TP-self-signed-2641177237
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-2641177237
 revocation-check none
 rsakeypair TP-self-signed-2641177237
crypto pki trustpoint SLA-TrustPoint
 revocation-check crl
crypto pki trustpoint router
 enrollment url http://10.30.1.153:80
```

```
 ip-address 10.30.1.114
 subject-name CN=vbranch
 vrf public-vrf
 revocation-check crl
 rsakeypair router
 auto-enroll regenerate
 hash sha512

crypto ikev2 authorization policy uCPE-author-pol
 pool uCPE-pool1
 dns 166.37.162.103 166.37.218.90
 pfs
 route set interface Loopback1001
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 fvrf public-vrf
 proposal uCPE-proposal


crypto ikev2 profile uCPE-profile
 description uCPE profile
 match fvrf public-vrf
 match identity remote email domain vpntest.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-author-pol
 virtual-template 1 mode auto


crypto ipsec transform-set tset_aes_256_sha512 esp-aes 256 esp-sha512-hmac
 mode tunnel
crypto ipsec df-bit clear
no crypto ipsec profile default
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 group14
 set ikev2-profile uCPE-profile
crypto call admission limit ike in-negotiation-sa 30



ip radius source-interface GigabitEthernet2

 route set interface Loopback1001

interface Loopback1

 ip vrf forwarding private-vrf
```

```
 ip address 90.90.90.1 255.255.255.255
interface Loopback2
 ip vrf forwarding private-vrf
 ip address 91.91.91.1 255.255.255.0
interface Loopback3
 ip vrf forwarding private-vrf
 ip address 90.90.90.201 255.255.255.128
interface Loopback1001
 description MNSO Interface and Tunnel Loopback
 ip vrf forwarding private-vrf
 ip address 166.35.121.112 255.255.255.255
interface GigabitEthernet1
 ip vrf forwarding private-vrf
 ip address 92.92.92.1 255.255.255.0
 negotiation auto
 no mop enabled
 no mop sysid
interface GigabitEthernet2
 ip vrf forwarding public-vrf
 ip address 10.30.1.114 255.255.255.0
 negotiation auto
 no mop enabled
 no mop sysid
interface GigabitEthernet3
 ip vrf forwarding private-vrf
 ip address 91.91.91.1 255.255.255.0
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
interface Virtual-Template1 type tunnel
 ip vrf forwarding private-vrf
```

```
 ip unnumbered Loopback1001

 ip mtu 1400

 ip tcp adjust-mss 1380

 tunnel vrf public-vrf

 tunnel protection ipsec profile uCPE-ips-prof

ip radius source-interface GigabitEthernet2
```

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 <vm_deploy_name>.<vm_group_name>
commit
```

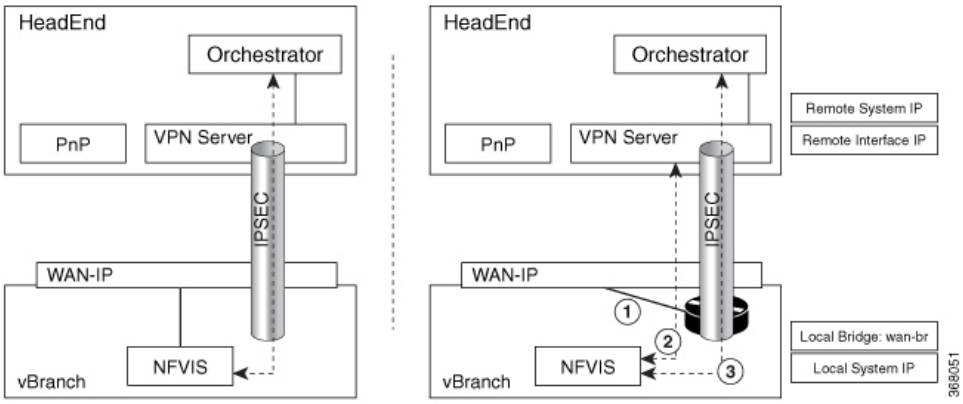
Use the **show vm_lifecycle deployments all | include Name** command to find out the Deployment Name and VM Group Name.

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 10: 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.

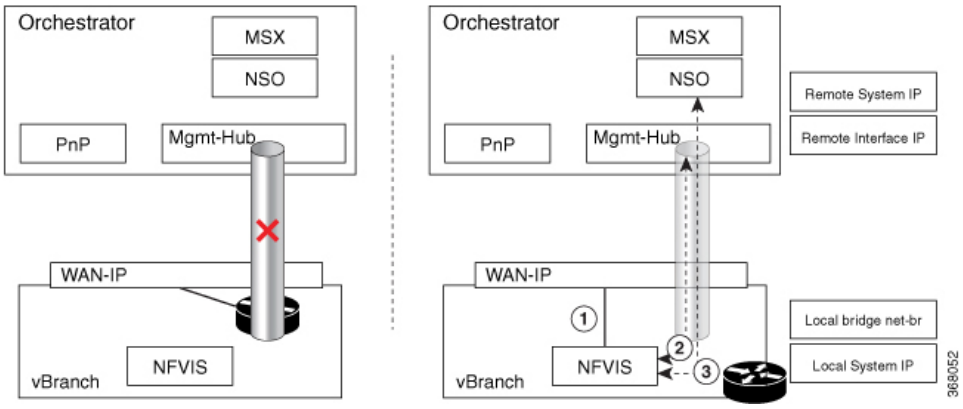
1. NFVIS deploys the VM with specified bootstrap and Day-0 configuration. NFVIS takes down the current IPSec tunnel and releases the public IP address.
2. The VM takes the public IP address when it is in active state. NFVIS sets up the IPSec tunnel again with the remote management hub.
3. After the IPSec 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 11: 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.

Each NFVIS has its int-mgmt-net and subnet associated to the IP address pool. The following is an exmaple to setup the IP address pool before deploying VM:

```
no vm_lifecycle networks network int-mgmt-net
commit
vm_lifecycle networks network int-mgmt-net
subnet int-mgmt-net-subnet address 12.12.12.0
netmask 255.255.255.0
gateway 12.12.12.1
dhcp false
ipversion ipv4
commit
```

Single IP Monitoring External Gateway

Before NFVIS 4.8, single IP monitors the external gateway using a ping, every 2 minutes, by default. Starting from NFVIS 4.8, singe IP monitors the external gateway only if you configure the **wan-monitor-interval** command. The monitor interval can range from 2 minutes to 60 minutes. If the **wan-monitor-interval** command is deleted, single IP stops monitoring the external gateway.

To configure the external gateway monitor interval:

```
nfvis(config)# single-ip-mode wan-monitor-interval ?
Description: Monitor interval in minute, value range from 2-60.
Possible completions:
  <short, 2 .. 60>
```

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.

The following is an example based on ISRv 16.09.01a that shows how to configure the guest router to route the traffic:

```
<?xml version="1.0" encoding="UTF-8"?>
<Environment
xmlns:oe="http://schemas.dmtf.org/ovf/environment/1">
  <PropertySection>
    <Property oe:key="com.cisco.csr1000v.config-version.1" oe:value="1.0"/>
    <Property oe:key="com.cisco.csr1000v.enable-ssh-server.1" oe:value="True"/>
    <Property oe:key="com.cisco.csr1000v.login-username.1" oe:value="${SSH_USERNAME}"/>
    <Property oe:key="com.cisco.csr1000v.login-password.1" oe:value="${SSH_PASSWORD}"/>
    <Property oe:key="com.cisco.csr1000v.mgmt-interface.1" oe:value="GigabitEthernet1"/>
!!!GigabitEthernet1-nicid(0)-int-mgmt-interface-don't change ip address or don't shutdown
    <Property oe:key="com.cisco.csr1000v.mgmt-ipv4-addr.1"
oe:value="${NICID_0_IP_ADDRESS}/24"/>
    <Property oe:key="com.cisco.csr1000v.mgmt-ipv4-network.1" oe:value=""/>
    <Property oe:key="com.cisco.csr1000v.license.1" oe:value="${TECH_PACKAGE}"/>
    <Property oe:key="com.cisco.csr1000v.ios-config-0001" oe:value="vrf definition
Mgmt-intf"/>
    <Property oe:key="com.cisco.csr1000v.ios-config-0002" oe:value="address-family ipv4"/>
    <Property oe:key="com.cisco.csr1000v.ios-config-0003" oe:value="exit-address-family"/>
    <Property oe:key="com.cisco.csr1000v.ios-config-0004" oe:value="address-family ipv6"/>
    <Property oe:key="com.cisco.csr1000v.ios-config-0005" oe:value="exit-address-family"/>
    <Property oe:key="com.cisco.csr1000v.ios-config-0006" oe:value="exit"/>
    <Property oe:key="com.cisco.csr1000v.ios-config-0007" oe:value="interface
GigabitEthernet1"/>
```

```
    <Property oe:key="com.cisco.csr1000v.ios-config-0008" oe:value="ip address
${NICID_0_IP_ADDRESS} ${NICID_0_NETMASK}"/>
    <Property oe:key="com.cisco.csr1000v.ios-config-0009" oe:value="no shut"/>
    <Property oe:key="com.cisco.csr1000v.ios-config-0010" oe:value="exit"/>
  </PropertySection>
</Environment>
```

The following is an example of ISRv day-0 configuration with a static public IP address:

```
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 route 10.19.18.0 255.255.255.0 12.12.12.1
!
ip access-list standard NAT
permit 12.12.12.0.0.0.0.25
```

The following is an example of ISRv day-0 configuration with DHCP IP address:

```
interface GigabitEthernet1
ip nat inside
negotiation auto
!
interface GigabitEthernet2
ip address dhcp
ip nat outside
negotiation auto
!
ip nat inside source list NAT interface GigabitEthernet2 overload
ip route 10.19.18.0 255.255.255.0 12.12.12.1
!
ip access-list standard NAT
permit 12.12.12.0.0.0.0.25
```

To verify single IP address mode status us the **show single-ip-mode** and **show secure-overlay** command:

```
nfvis#

single-ip-mode state start
single-ip-mode state-details "waiting for VM ROUTER.ROUTER deployment"


nfvis#

single-ip-mode state start
single-ip-mode state-details "VM deployed"
nfvis#
nfvis# show secure-overlay

                   ACTIVE LOCAL      STATE
NAME    STATE      BRIDGE           DETAILS
-----------------------------------------------------
mgmthub  up      int-mgmt-net-br   VM alive
```

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, and Single-IP works only on the GE0-0 port.

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.

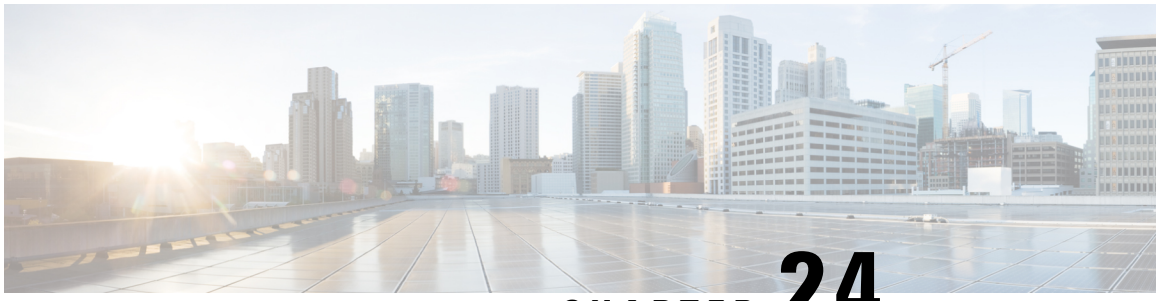
The following example is a sample bootstrapping configuration:

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 24

Dual WAN Support

Dual WAN support provides 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.

- Bridge IP Configurations, on page 245
- Dual WAN Bridge and DHCP Toggle, on page 246

Bridge IP Configurations

Both default bridges and user generated bridges contain IP/DHCP configuration to allow IP configurations on any port. NFVIS provides two ports by default for zero touch deployment, with dhclient actively requesting DHCP IP configurations.

Each bridge can be configured with:

- IPv4 DHCP
- Static IPv4
- IPv6 DHCP
- IPv6 SLAAC
- Static IPv6
- VLAN tag

Except for ENCS 5400 management port configuration which continues to remain under **system settings mgmt**, use the Bridge APIs and Commands to enable IP configurations and move away from the previous System Settings APIs and commands.

On ENCS5100 use **system settings mgmt** for management IP configuration and **bridges bridge lan-br** for LAN IP configuration.

Restrictions for Bridge IP Configurations

IPv4:

- IPv4 DHCP can only be configured on one bridge at a time
- Cannot configure IPv4 DHCP on any bridge if system wide default gateway is configured
- Cannot configure **system settings default-gw** if IPv4 DHCP is configured on any bridge.

IPv6:

- IPv6 DHCP can only be configured on one bridge at any time, and cannot be applied on any bridge if IPv6 SLAAC is applied on any bridge or if system wide IPv6 default gateway is configured.
- IPv6 SLAAC can only be configured on one bridge at any time, and cannot be applied on any bridge if IPv6 DHCP is applied on any bridge or if system wide IPv6 default gateway is configured.
- Cannot configure **system settings default-gw-ipv6** if IPv6 DHCP or IPv6 SLAAC is configured on any bridge.

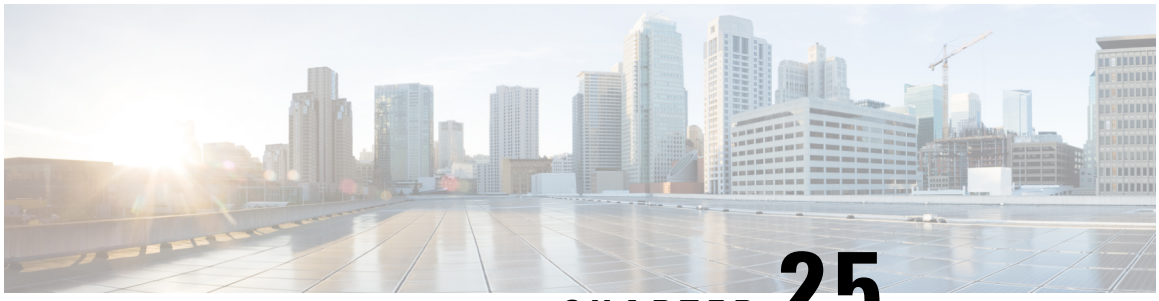
Dual WAN Bridge and DHCP Toggle

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 a static IP address 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 and it is not possible to toggle between WAN bridges. 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.



CHAPTER 25

Switch Port Security

- Switch Port Security, on page 247

Switch Port Security



Note Always shutdown interface before port security configurations.

Port security is not supported on port channel interfaces.

Restrictions

- Dynamic secure mac address are not retained over reboot and only delete-on-reset secure mode is supported.
- Static mac address must be set after port is in shutdown, and other port security commands are configured and enabled.
- Only ethernet ports are supported for port security configuration.
- Does not support trap and **show switch interface port-security** command does not show information about trap.
- If port-security violation shutdown mode is configured on a secure port and violation traffic is received, the port changes to error state. A manual interface shutdown and no shutdown is required to recover the port.

Configuring Port Security

To configure port security:

1. Shutdown the interface:

```
configure terminal
switch
    interface gigabitEthernet 1/1
    shutdown
    commit
```

2. Disable port security:

```
configure terminal
switch
    interface gigabitEthernet 1/1
    no port-security enable
    commit
```

3. Configure max mac address:

```
configure terminal
switch
    interface gigabitEthernet 1/1
    port-security max 5
    commit
```

4. Configure violation handling:

```
configure terminal
switch
    interface gigabitEthernet 1/1
    port-security violation discard
    commit
```

5. Enable port security:

```
configure terminal
switch
    interface gigabitEthernet 1/1
    port-security enable
    commit
```

6. Add static secure mac address:

```
configure terminal
switch
    mac address-table static 18:65:90:cb:e6:08 vlan 1 interface gigabitEthernet 1/1
secure
    commit
```

7. Restart the disabled interface:

```
configure terminal
switch
    interface gigabitEthernet 1/1
    no shutdown
    commit
```

8. Use **show switch interface port-security** to verify the configuration:

```
nfvis# show switch interface port-security
MAC
VIOLATION ADDRESS MAX MAC
PORT STATUS LEARNING HANDLING COUNT ADDRESS
-------------------------------------------------------------
1/0 Disabled Delete-On-Reset Discard 0 0
1/1 Enabled Delete-On-Reset Discard 1 5
1/2 Disabled Delete-On-Reset Discard 0 0
1/3 Disabled Delete-On-Reset Discard 0 0
```

```
1/4 Disabled Delete-On-Reset Discard 0 0
1/5 Disabled Delete-On-Reset Discard 0 0
1/6 Disabled Delete-On-Reset Discard 0 0
1/7 Disabled Delete-On-Reset Discard 0 0
```

9. Use **show switch mac addr-table** command to check static configured or dynamic learnt secure mac addresses:

```
nfvis# show switch mac addr-table
VLAN MAC ADDRESS PORT TYPE
---------------------------------------
1 18:65:90:cb:e6:08 gi1/1 secure
```



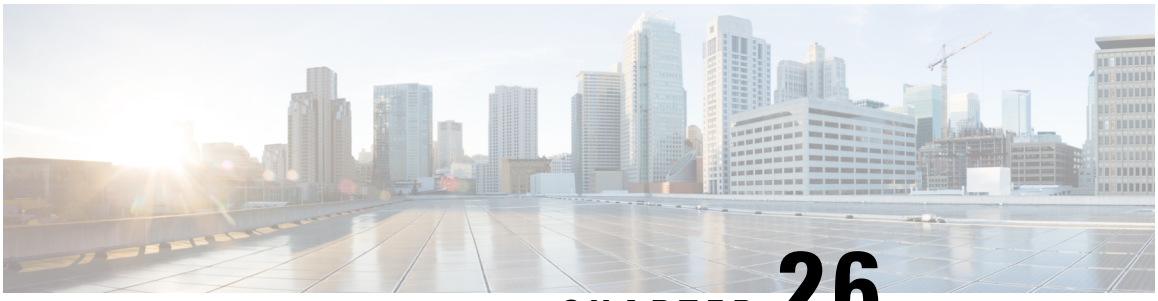
Note If traffic with a secure MAC address that is configured on one secure port attempts to access another secure port in the same VLAN, ENCS switch port security responds to the violation by discarding the traffic always.



PART II

NFVIS Functionality Changes for Cisco SD-WAN Cloud OnRamp for Colocation

- Default System Configurations, on page 253
- NFVIS Integration with Docker Container Lifecycle, on page 257
- NFVIS Integration with vManage, on page 263
- Enhancements to VM Image Packaging, on page 267



CHAPTER 26

Default System Configurations

LACP

In Cisco SD-WAN Cloud OnRamp for Colocation solution, the Link Aggregation Control Protocol (LACP) is enabled for the management port channel. The management port channel is created by default using the Ethernet links (eth0-1 and eth0-2). To ensure the port channel configuration on the management switch side is reachable, run the **support ovs appctl bond-show mgmt-bond** command and ensure OOB switch ports that are connected to the switch has the following port-channel configuration.

```
!
interface Port-channel1
 switchport mode access
!
interface GigabitEthernet1/0/6
 switchport mode access
 channel-group 1 mode passive
!
interface GigabitEthernet1/0/7
 switchport mode access
 channel-group 1 mode passive
!
```

DHCP

In Cisco SD-WAN Cloud OnRamp for Colocation solution, DHCP is enabled by default on the management port channel. Once the DHCP server is up, the host gets the DHCP IP address on management internal port.

Sticky DHCP

Note Sticky DHCP configurations are optional.

Configure the DHCP servers to get sticky DHCP IP address. The DHCP client identifier is the serial number on the CSP device.

The DHCP server configuration on a Linux server is:

```
host
{
option dhcp-client-identifier "WZP22060AUR";
fixed-address 10.20.0.2;
```

```
option routers 10.20.0.1;
option domain-name-servers 198.51.100.9;
option domain-name "cisco.com";
option subnet-mask 255.255.0.0;
}
```

The DHCP server configuration on IOS is:

```
ip dhcp pool P_112
host 10.0.0.2 255.255.0.0
client-identifier 4643.4832.3133.3256.3131.48
default-router 10.0.0.1
dns-server 10.0.0.1
```

Here 10.0.0.2 is the sticky DHCP IP address. Use **debug ip dhcp server packet** command to find out the client identifier.

Static IPv4

To troubleshoot issues with DHCP configurations, configure static IPv4 on the management port channel:

```
configure shared
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
system settings domain cisco.com
system:system settings dns-server 209.165.201.20
system:system settings ip-receive-acl 0.0.0.0/0
action accept
priority 100
service scpd
commit
```

Since vManage is the controller in this solution, **Configure shared** writes to candidate database (CDB) which will keep the device config in sync with vManage.

Note **Configure shared** is only applicable to static ip configurations. Any other configurations done manually using either confd cli or netconf or rest api, will be removed by vManage as NFVIS is a vManaged device in this solution.

In NFVIS NetworkHub image, networks are automated and user should not create, delete or modify networks. You can reset the host server to default configurations using the **factory reset all** command.

- SRIOV Support, on page 254
- OVS-DPDK Support, on page 255

SRIOV Support

SR-IOV is statically enabled on NFVIS Cisco SD-WAN Cloud OnRamp for Colocation image with a CSP 5444 Product Identifier (PID).

- SRIOV is enabled by default on ethernet ports eth1-1, eth1-2, eth4-1, eth4-2 as Niantec NIM cards are placed in slots 1 and 4.
- SR-IOV is enabled only on Niantic NICs and onboard Niantics does not support SR-IOV.

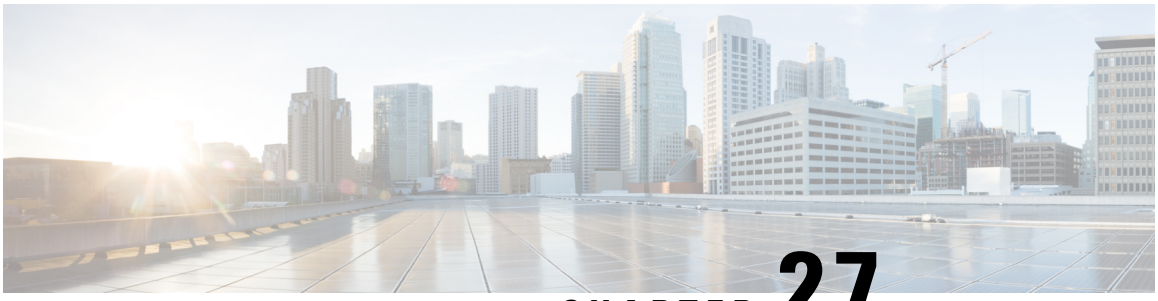
- Thirty two virtual functions are created on each PNIC. . If the NIC is connected to 1G, two virtual functions are created.
- Virtual Ethernet Port Aggregator (VEPA) mode is enabled.
- The naming convention is: <interface name>-SRIOV-1,<interface name>-SRIOV-2 ,<interface name>-SRIOV-3,<interface name>-SRIOV-4.
- Fortville NICs are used to create port channels for OVS data traffic and HA sync between the VM's.

OVS-DPDK Support

Starting from NFVIS 3.12 release, OVS-DPDK support is enabled on NFVIS. The data and HA bridges are ovs-dpdk bridges by default. The bridges are associated with bonds created over pnics on the Fortville card. The pnics are bound to DPDK compatible drivers. OVS-DPDK support provides higher performance than the standard kernel OVS datapath.

NFVIS System has two CPUs from the same socket. For each socket one CPU is reserved for DPDK. The kernel is allocated 8GB of memory. 4G memory is allocated per socket for the DPDK poll mode driver. The rest of the memory is converted to 2MB huge pages and allocated for VM deployment.

To check the OVS-DPDK status, use **Show system:system settings dpdk-status** command.



CHAPTER 27

NFVIS Integration with Docker Container Lifecycle

Docker container lifecycle infrastructure is developed in NFVIS for Cisco SD-WAN Cloud OnRamp for Colocation solution. Container lifecycle APIs are developed to bring up docker containers.

- Cisco Colo Manager, on page 257

Cisco Colo Manager

Cisco colo manager (CCM) is a software stack managing switches in colo. In the Cisco SD-WAN Cloud OnRamp for Colocation solution, CCM is hosted on NFVIS software in a Docker container. CCM is hosted on the CSP devices along with VNFs and there are no dedicated CSP devices for hosting CCM. CCM is used to configure and provision PNFs (switches) in this solution.

Cisco colo manager (CCM) is bundled along with the Cisco NFVIS software which is used as the base virtualization infrastructure software running on the compute platform. The NFVIS software provides programmable Rest and netconf APIs and an orchestrator can use these APIs to configure and monitor the system, instantiate virtual network functions and configure the VNF networks and service chains. As part of colo provisioning for the orchestrator, vManage selects one device in the colo and sends netconf action command to bring up the CCM container. The CCM container is connected to the colo management network. This management network is used to transfer files and images into and out of the systems. This network will not be used for the normal customer data traffic.

CCM State Transitions from the Host Side

vManage brings up CCM on one of the CSP devices in the Cloud OnRamp for Colocation solution. CCM state transitions are seen on the host side, using the container life-cycle model's state operation.

The CCM state on the host side has the following states:

- Starting : when CCM has been brought up and health check script has not been run. During this phase, vManage waits for CSP state to change to Healthy.
- Healthy : when the health check script has been run and it has passed the checks. This state implies that the operational model for configuration status can be queried or configuration can be pushed. During this phase, if CCM is in INIT state, vManage pushes the device list. If CCM is not in INIT state, Cloud OnRamp for Colocation may be in degraded state and recovery flow must happen.

- Unhealthy : If the CCM does not boot properly, the CCM container is not usable and needs to be recovered. CCM in unhealthy state can be due to docker daemon not running, CCM is not configured with correct management IP address, gateway or CCM cannot respond to ping.

The starting state can only be seen when the container is brought up or re-spun. Healthy and unhealthy states can transition to each other during the lifetime of the container. A notification is also sent whenever the CCM state changes.



Note The CCM container state is tracked through container life-cycle model as one of the containers. This is not CCM-state or CCM-status oper. The state for container named ColoMgr is used for CCM state transitions.

State	Action/config can be pushed	config status queried	oper model on host	notification for CCM state
Starting	No	No	Yes	Yes
Unhealthy	No	No	Yes	Yes
Healthy	Yes	Yes	Yes	Yes

PNF device list is sent from vManage to the NFVIS hosting CCM when CCM is in healthy state.

To verify CCM state, when Colo Manager crashes on a CSP device use **support show container** command:

```
CSP# support show container
Possible completions:
  docker-container-ls   Lists all containers
  docker-info           Lists docker daemon info
  docker-inspect        Inspect container or volume
  docker-volume-ls      Lists all volumes
  dump                  Dumps all container related info
```

CCM Notifications

CCM health check sends CCM state transitions to vManage notification stream.

You can view the CCM event notifications using the **show notification stream vmanageEvent** command.

Event Type	Notification Trigger	Notification Output Example
ccmEvent - CCM-STATUS (init, in-progress, success, failure)		notification eventTime 2018-06-29T01:58:55.767142+00:00 ccmEvent severity-level minor host-name ccm user-id nso_user config-change false transaction-id 0 status SUCCESS status-code 0 status-message INIT details CCM status :INIT event-type CCM-STATUS !

CCM Recovery

When CCM is up, the Catalyst 9000 series switches are onboard successfully and CCM is restarted on the same or different CSP, the CCM recovery is initiated.

vManage brings down CCM and then brings it up again. vManage sends the device list with passwords for the switches along with all the service configurations. CCM then uses these configurations to sync with the device.

Recovery flag for device action list - false for day0, true for recovery (mandatory).

Static IP change for device action list - IP addresses for devices is sent all the time - day0 and recovery.

Support Commands

To verify the CCM version use **support show ccm-version**:

```
CSP# support show ccm-version
Cisco Colo Manager (CCM)
Version 0.0.1-150
Build date Tue 06 Nov 2018 09:09:28 AM UTC
```

To verify the firewall state use **support show firewall**:

```
CSP# support show firewall
```

```
Possible completions:
  list-forward-ports   Lists all port forwarding rules
  state                Lists firewalld daemon status
```

To display information about OVS switch use **support ovs vsctl show**:

```
CSP# support ovs vsctl show
Possible completions:
  |  <cr>
CSP2# support ovs vsctl show
107a6588-62f1-411f-b5da-fa0fd39f2500
    Bridge ovs-data-br
        Port bond-bond_data
            tag: 1
            Interface "eth2-3"
            Interface "eth2-4"
        Port ovs-data-br
            Interface ovs-data-br
                type: internal
    Bridge ovs-ha-br
        Port bond-bond_ha
            tag: 1
            Interface "eth2-2"
            Interface "eth2-1"
        Port ovs-ha-br
            Interface ovs-ha-br
                type: internal
    Bridge int-mgmt-net-br
        Port colo-mgmt
            Interface colo-mgmt
                type: internal
        Port mgmt-bond
            Interface "eth0-2"
            Interface "eth0-1"
        Port int-mgmt-net-br
            Interface int-mgmt-net-br
                type: internal
    ovs_version: "2.5.2"
```

To display the list of NFVIS system settings use **show system:system settings-native**:

```
system:system settings-native mgmt ip-info interface colo-mgmt
system:system settings-native mgmt ip-info ipv4_address 192.168.30.163
system:system settings-native mgmt ip-info netmask 255.255.255.0
system:system settings-native mgmt ip-info link-local ipv6 address ::
system:system settings-native mgmt ip-info link-local ipv6 prefixlen 0
system:system settings-native mgmt ip-info global ipv6 address ::
system:system settings-native mgmt ip-info global ipv6 prefixlen 0
system:system settings-native mgmt ip-info mac_address b2:5d:28:aa:f1:96
system:system settings-native mgmt ip-info mtu 1500
system:system settings-native mgmt ip-info txqueuelen 1000
system:system settings-native mgmt stats rx_packets 7140693
system:system settings-native mgmt stats rx_bytes 767558248
system:system settings-native mgmt stats rx_errors 0
system:system settings-native mgmt stats rx_dropped 2
system:system settings-native mgmt stats rx_overruns 0
system:system settings-native mgmt stats rx_frame 0
system:system settings-native mgmt stats tx_packets 5259073
system:system settings-native mgmt stats tx_bytes 1008512311
system:system settings-native mgmt stats tx_errors 0
system:system settings-native mgmt stats tx_dropped 0
system:system settings-native mgmt stats tx_overruns 0
system:system settings-native mgmt stats tx_carrier 0
```

```
system:system settings-native mgmt stats tx_collisions 0
system:system settings-native domain NA
system:system settings-native dns nameserver1 0.0.0.0
system:system settings-native dns nameserver2 0.0.0.0
system:system settings-native dns nameserver3 0.0.0.0
system:system settings-native hostname CSP2
system:system settings-native gateway ipv4_address 192.168.30.1
system:system settings-native gateway interface colo-mgmt
system:system settings-native gateway-ipv6 ipv6_address ::
system:system settings-native gateway-ipv6 interface NA
system:system settings-native trusted-source [ "not set" ]
system:system settings-native source-interface 0.0.0.0
```

To display information about a bond use **support ovs appctl bond-show mgmt-bond**

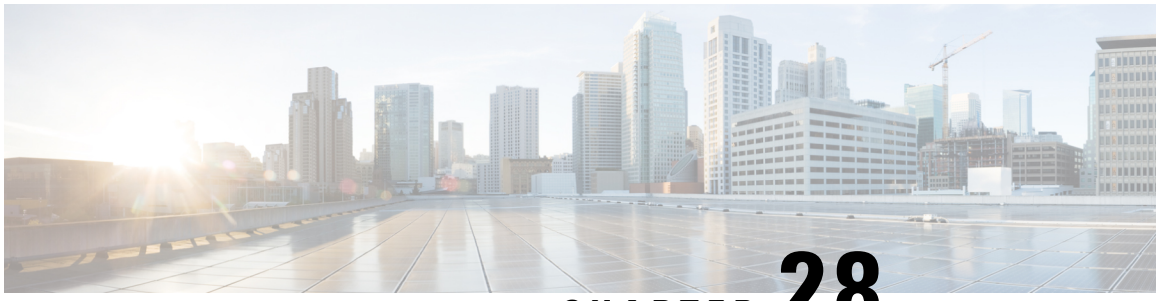
```
CSP2# support ovs appctl bond-show mgmt-bond
---- mgmt-bond ----
bond_mode: balance-slb
bond may use recirculation: no, Recirc-ID : -1
bond-hash-basis: 0
updelay: 0 ms
downdelay: 0 ms
next rebalance: 252 ms
lacp_status: negotiated
active slave mac: 00:fc:ba:d7:39:86(eth0-1)

slave eth0-1: enabled
        active slave
        may_enable: true
        hash 242: 8 kB load

slave eth0-2: enabled
        may_enable: true
```

To display the IP routing statistics use **support show route**:

```
CSP# support show route
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         gateway         0.0.0.0         UG        0 0          0 colo-mgmt
172.16.255.22   127.0.1.254     255.255.255.255 UGH       0 0          0 tun_0_0
192.168.30.0    0.0.0.0         255.255.255.0   U         0 0          0 colo-mgmt
```



CHAPTER 28

NFVIS Integration with vManage

In the Cisco SD-WAN Cloud OnRamp for Colocation solution a colo is a stack of computing and networking fabric which brings up multiple networking functions and service chains them to connect branch users or endpoints to hybrid cloud or data center. vManage is used as the orchestrator to provision the devices in a colo. This solution can be deployed in multiple locations where each colo is independent and unaware of other colos in the same site or across sites.

- Establishing DTLS Tunnel with vManage, on page 263
- NFVIS Notifications, on page 265
- Stats for Host and VM, on page 265
- System CLI, on page 265
- NFVIS Local Portal, on page 265
- Core Allocation for Host and CCM, on page 265

Establishing DTLS Tunnel with vManage

Before you begin

To establish a DTLS channel with vManage, vDaemon is integrated on NFVIS

Note The device is vManaged and hence any configurations done out-of-band is overwritten by vManage. The show commands continue to work in the same way.

Note If CSP devices are already added into PnP Connect, skip the instructions that are mentioned from steps 1 to 5 in topic, and perform instructions from step 6.

Note If CSP devices are already added into vManage, perform instructions from step 13.

Procedure

Step 1 Get access to PnP devices and log into Plug and Play Connect.

Step 2 Create a virtual account. See the Plug and Play Connect Configuration Guide for more information about creating a virtual account.

Step 3 In the virtual account, create a vbond controller.

Note
Only one vbond controller profile is allowed in a virtual account.

Step 4 In the Add Controller Profile window, provide information about Organization Name, vbond IP address, root CA, and other information. Click **Next**.

Step 5 Go to the **Devices** tab, add your device by using PID and serial number. Assign the vbond profile that is created in step 3 to the device.

Note
You can only choose and add CSP 5444, X1 and X2 devices.

If the switch and CSP devices are already added into PnP Connect, skip steps 1 to 5. Go to the next step.

Step 6 Your device should have DNS servers with connections to Plug and Play Connect.

Step 7 Verify PnP status to determine if redirection is successful. Use the `nfvis# show pnp status` command to determine PnP status.

Step 8 Go to Plug and Play Connect screen and verify if status is displayed as "Redirect Successful".

Step 9 To ensure that VPN configuration are present on NFVIS, use the `nfvis# show running-config vpn` command.

Step 10 To ensure that Organization name and vbond IP address have been configured, use the `nfvis# show running-config viptela-system:system` command.

Step 11 To ensure that root ca have been installed, use the `nfvis# show control local-properties root-ca-chain-status` command.
If the switch and CSP devices are already added into vManage, skip the next step and perform instructions from step 13.

Step 12 Upload WAN edge list into vManage. For more information, see Add Cloud OnRamp for Colocation Devices into vManage in Cisco SD-WAN Cloud OnRamp for Colocation Solution Guide.

Step 13 In vManage, go to **Configuration** > **Network Hub** screen. Create a new cluster by clicking the **Configure & Provision Cluster** button. For more information, see Create and Activate Network Hub Cluster in Cisco SD-WAN Cloud OnRamp for Colocation Solution Guide.

Step 14 After activating the cluster, get the token that you had noted while adding devices into vManage, and then request an activate command on NFVIS. Use the following NFVIS command:

Example:

```
nfvis# request activate chassis-number CSP-5444-X2-FCH2118V0CY token
f3117c35c3206f4adfab5ced0d57db44
```

Step 15 Verify that your system IP address has been configured, VNFs to be run on CSPs such as CSR 1000v, vEdge are already installed, and connections are working. For verification, use the following NFVIS commands:

Example:

```
nfvis# show control local-properties certificate-status
nfvis# show control local-properties system-ip
nfvis# show control connections
```

Step 16 If connections are not running, use the following NFVIS history command to debug:

Example:

```
nfvis# show control connections-history
```

NFVIS Notifications

You can view the NFVIS notifications using the **show notification stream viptela** command. The NFVIS notifications are available at Syslog Messages, on page 173 and the same notifications are sent to viptela stream.

Stats for Host and VM

The stats for cpu/mem/disk/interface are collected periodically and the files are compressed and stored in the device in the required format for vManage. vManage collectes these log files periodically and deletes the older set of log files.

System CLI

In NFVIS integration with vManage, Viptela system model is loaded into NFVIS software, due to which the existing NFVIS **system** commands become **system:system** commands.

Example:

```
show system:system status
```

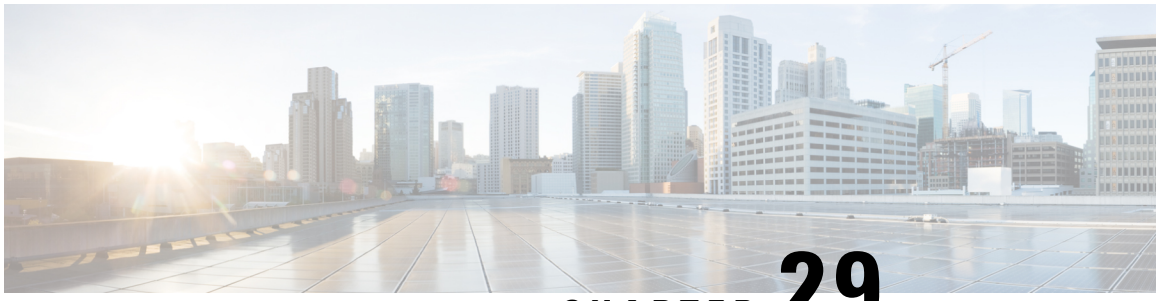
NFVIS Local Portal

All the configurations from the local portal are blocked as the admin role is changed to view-only for Cisco SD-WAN Cloud OnRamp for Colocation solution. The admin can continue to use the NFVIS portal for troubleshooting and serviceability, but not for provisioning or configuring any functionalities.

Core Allocation for Host and CCM

The host CPU reserve based on the hardware core is :

- less than 12 cores : 1 pCPU for NFVIS 1pCPU=1core
- 16 cores : 2 pCPUs for NFVIS
- greater than 16 cores : 4 pCPUs for NFVIS



CHAPTER 29

Enhancements to VM Image Packaging

The Cisco Enterprise NFVIS VM image packaging tool, nfvpt.py, is enhanced to support functionality required for Cisco SD-WAN Cloud OnRamp for Colocation solution.

- NFVIS Specific Enhancements, on page 267
- Cisco SD-WAN Cloud OnRamp for Colocation Packaging Enhancements, on page 268
- VM Packaging Parameters, on page 269
- VM Packaging Utility Usage Examples, on page 270
- Packaging a VM, on page 270

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

Cisco SD-WAN Cloud OnRamp for Colocation Packaging Enhancements

The following parameters are the enhancements specific to SD-WAN:

```
--json JSON           Provide JSON input for bootstrap variables; mutually
                        exclusive with custom and bootstrap configs
  --multi_use           Add options for use in multiple use-cases
  --app_vendor APP_VENDOR
                        Application Vendor e.g. Cisco, Juniper etc

  --bootstrap BOOTSTRAP
                        Every bootstrap file should be a different option HA
                        packaging format: --bootstrap mount_point:<value>,file
                        :<file2mount>[,<attrib>:<value>] mount_point:<value>
                        and file:<file2mount> are mandatory followed by one or
                        more attributes in the format <attrib>:<value> Legacy
                        format: --bootstrap file1,file2... See usage.txt for
                        more details


  --ha_package          enable HA packaging
  --mgmt_vnic MGMT_VNIC
                        VM management interface identifier
HA options:
  --ha_capable
  --ha_vnic HA_VNIC     VM HA vnic CSV list

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
```

VM Packaging Parameters

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

Parameter	Mandatory/Optional	Description
json	Optional	Provide JSON input for bootstrap variables. It's mutually exclusive with custom and bootstrap configs
multi_use	Optional	option for use in multiple use-cases
ha_package	Optional	enable HA packaging
mgmt_vnic	Optional	VM management interface identifier
pack_dir	Optional	package all files in directory
app_vendor	Required	Application Vendor e.g. Cisco, Juniper
ha_capable	Optional	For HA capability
vnic_names	Optional	list of vnic number to name mapping in format number:name --vnic_names 1:GigabitEthernet2,2:GigabitEthernet4

VM Packaging Utility Usage Examples

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

Example 1: 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 1: 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 1: 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
```

Packaging a VM

The following steps shows how to package a bundled VM image, bootstrap files and metadata into an archive:

1. Create a json file using gen_json.py tool. The gen_json.py needs a pattern that matches bootstrap files as an option. gen_json.py --help shows all the details about the options. Redirect the output of gen_json.py into a json file.

   ```
   gen_json.py --g "boot*,ios*" --ha > temp.json
   ```

   Include --ha option if the packaging is for HA. Include --multi_use option if the VM is a part of a service chain.

2. The temp.json file has two arrays - Userinput and SysGen. Userinput and SysGen are variables from bootstrap files which were tokenized. By default all the variables are included in Userinput array. The system generated variables should be moved to SysGen array. vManage generates some of these variables like MGMT and DATA IP addresses from the pool provided in the cluster creation on vManage. All other variables like DNS_SERVER, VM password etc. are user inputs at the VM/servicechain provisioning.

   Example:

   ```
   interface G0/1
   ip address ${MGMT_PRIM} <-- variable
   ```

3. After making changes to the json file you can package the VM with the script - nfvpt.py.

   ```
   nfvpt.py -i <qcow file> -o <tar file name> --json  <json file>  --app_vendor cisco -t
   firewall -r 10 --optimize true -n asav --monitored true --ha_package —ha_capable
   ```

   The tool creates a .tar.gz file with the name you have provided.