

Cisco Enterprise Network Function Virtualization Infrastructure Software Configuration Guide, Release 4.x

First Published: 2020-03-31

Last Modified: 2025-12-19

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)

© 2020–2024 Cisco Systems, Inc. All rights reserved.



CONTENTS

Full Cisco Trademarks with Software License ?

CHAPTER 1 New and Changed Information 1

CHAPTER 2 About NFVIS Documentation 9

CHAPTER 3 Host System Management 11

- System Access Configuration 12
  - Host System Requirements 12
  - System Setting Hostname 14
  - Dual WAN Support 15
    - Restrictions for Dual WAN Support 15
    - Dual WAN Bridge and DHCP Toggle 15
  - Accessing NFVIS 15
  - Configuring VLAN for NFVIS Management Traffic 21
  - Configuring the IP Receive ACL 21
    - Port 22222 and Management Interface ACL 21
  - Configuring Secondary IP Address and Source Interface 22
  - CIMC Access Control 23
    - CIMC Access using NFVIS 23
    - BIOS-CIMC Update 24
    - BIOS and CIMC Password 24
    - UEFI Secure Boot on ENCS 5400 25
    - PXE Boot Mode 26
  - Enable or Disable Access to NFVIS Portal 26
- Users, Roles, and Authentication 27

Local User Account Management 27
User and Role Management in the NFVIS Portal 31
Create New Users 31
Modify Users 32
Delete Users 33
Change Language Preferences in the NFVIS Portal 33
User Groups 34
Granular Role-Based Access Control 34
Local Authentication for a Specific Group of Users 37
RADIUS Support 38
About RADIUS 38
RADIUS Operation 38
Configuring RADIUS 38
TACACS+ Support 39
About TACACS+ 39
TACACS Operation 40
Configuring a TACACS+ Server 40
Default Authentication Order 42
Networking 43
Bridges 43
Creating Bridges 44
Configuring Bridge Port 44
Configuring Bridge IP Connectivity 45
Physical Network Interface Cards 49
Configuring LLDP 49
Configuring Administrative Status of a Port 50
Tracking Changes for a Port 51
Speed, Duplex and Autonegotiation 52
Port Channels 54
Promiscuous mode 57
Dynamic SR-IOV 58
System Routes 59
Configuring System Routes 59
Troubleshooting 60

Cisco Network Plug-n-Play Support 60
PnP Discovery Methods 62
Configuring PnP Discovery Methods 62
Certificate for Static PnP discovery 66
PnP Action 68
DPDK Support on NFVIS 68
Storage Access 71
Network File System Support 71
Host System Operations 71
Backup and Restore NFVIS and VM Configurations 75
APC UPS Support and Monitoring 83
Resetting to Factory Default 84
Configure Banner, Message of the day and System Time 85
Configuring Your Banner and Message of the Day 85
Setting the System Time Manually or With NTP 85
Configure DNS Name Servers 87
Configuring IP Host 88

CHAPTER 4 **VM Life Cycle Management** 89
Overview of VM Life Cycle Management 90
Workflow of VM Life Cycle Management 90
Support for NFVIS Container Lifecycle Management 91
Uploading VM Images to an NFVIS Server 92
Performing Resource Verification 93
Configuring Management IP Subnet 94
VM Image Packaging 94
VM Image Packaging Utility 94
Contents 94
Usage 95
NFVIS Specific Enhancements 101
VM Packaging Utility Usage Examples 101
Standard VM Image Packaging 103
Generating a VM Package 103
Appendix 104

VM Image Package Files 104
Image Properties Template File 112
Image Registration 113
Register VM Packages Using REST API 114
Register VM Image with Multiple Root Disks 115
Register a VM Image through a Root Disk 115
Register a Remote VM Image 116
Specify Storage Location for a VM Image 117
Update VM Image 117
Image Properties 118
NFVIS Container Image Registration 127
Image Registration for CUCM Applications 127
VM Profiles or Flavors 130
Example: Create VM Profile Using Rest API 130
Configure Internal Management Network 130
VM Deployment and Management 133
VM Deployment 133
Example: Deploy VMs Using REST API 133
VM Deployment Parameters 135
VM Bootstrap Configuration Options with a VM Deployment 139
AutoVF Support 140
VM Monitoring 140
VNF Deployment Placement 141
VM Export with Selective Disk 142
VNF Volumes 142
Port Forwarding 144
NGIO 144
VNF Deployment Using Granular RBAC 145
Deploy CUCM Application 146
Recovery Procedures for CUCM Applications 147
VM States 149
NFVIS Container Deployment 150
VNF Deployment Update 152
Update VNF Flavor 152

Update CPU Topology 153
About Updating VNF Interfaces 154
Access VNFs 157
Access VNFs Using VNC Console 157
Access VMs Using Serial Console 158
Access a VM Using Port Forwarding 158
Import and Export NFVIS VM 159
Secure Boot of VNFs 162
CDROM Attachment and Detachment 163
Information about CDROM Attachment and Detachment 163
Prerequisites for CDROM Attachment and Detachment 163
Manage CDROM Attachment and Detachment 163
VM Graceful Stop 164
Information about Graceful VM Stop 164
Perform a Graceful stop 164

CHAPTER 5 **Cisco NFVIS Kubernetes Support 167**
Information About Cisco NFVIS Kubernetes Support 167
Prerequisites for Cisco NFVIS Kubernetes Support 168
Restrictions For Cisco NFVIS Kubernetes Support 168
Deploy Kubernetes on Cisco NFVIS 168
Deploy Cisco Kubernetes Using Cisco NFVIS 168
Verify Cisco NFVIS Kubernetes Deployment 170
Verify Kubernetes Deployment using CLI 171
Access Kubernetes Cluster 172
Troubleshoot Cisco NFVIS Kubernetes VM Support 173

CHAPTER 6 **Cisco NFVIS ThousandEyes Support 175**
Information About Cisco NFVIS ThousandEyes Support 176
Prerequisities For Cisco NFVIS ThousandEyes Support 176
Restrictions For Cisco NFVIS ThousandEyes Support 176
Deploy ThousandEyes on Cisco NFVIS 176
Deploy ThousandEyes Container from NFVIS Web Interface 177
Deploy ThousandEyes Container on NFVIS Using CLI Configs 178

Access ThousandEyes Using Cisco NFVIS 186

CHAPTER 7 **Cisco NFVIS Support For Cisco UCS C-Series Rack Servers 187**

Information About Cisco NFVIS Support 188

Supported Devices For Cisco NFVIS Support For Cisco UCS Rack Servers 188

Prerequisites For Cisco NFVIS Support For Cisco UCS C-Series Rack Servers 189

Restrictions For Cisco NFVIS Support For Cisco UCS C-Series Rack Servers 189

Configure Cisco Smart Licensing for Cisco NFVIS 189

Smart Mode 189

CSLU Mode 191

Monitor Cisco NFVIS Support for Cisco UCS C-Series Rack Server 191

Action Commands 194

CHAPTER 8 **Secure Overlay and Single IP Configuration 197**

Secure Overlay 197

Single Public IP Address and Secure Overlay 207

Single IP Address Without Secure Overlay 212

CHAPTER 9 **BGP Support on NFVIS 215**

Configure BGP on NFVIS 215

Route Distribution 218

BGP Route Annoucement over MPLS or IPSec 219

CHAPTER 10 **Security Considerations 223**

Installation 224

Image Tamper Protection 224

RPM Signing 224

RPM Signature Verification 224

Image Integrity Verification 224

ENCS Secure Boot 225

Secure Unique Device Identification 225

Device Access 226

Enforced Password Change at First Login 227

Restricting Login Vulnerabilities 227

Enforcement of Strong password 227
Configuring Minimum Length for Passwords 227
Configuring Password Lifetime 227
Limit previous password reuse 228
Restrict Frequency of login attempts 228
Disable inactive user accounts 228
Enforce Setting of BIOS and CIMC Passwords 229
Restrictions for Enforcing Setting of BIOS and CIMC Passwords 229
Information About Enforcing Setting of BIOS and CIMC Passwords 229
Configuration Examples for Enforced Resetting of BIOS and CIMC Passwords 230
Verify BIOS and CIMC Passwords 231
Integration with external AAA servers 231
Authentication Cache for External Authentication Server 231
Role Based Access Control 232
Restrict Device Accessibility 234
Attack vector reduction 234
Enabling only essential ports by default 234
Restrict Access To Authorized Networks For Authorized Services 235
Privileged Debug Access 236
Secure Interfaces 237
Console 237
SSH 237
CLI Session timeout 238
NETCONF 238
REST API 238
NFVIS Web Portal 238
HTTPS 239
SNMP Access 241
Legal Notification Banners 242
Factory Default Reset 243
Infrastructure Management Network 244
Out-of-band Management 244
Pseudo out-of-band Management 245
In-band Management 245

Locally Stored Information Protection 245
Protecting Sensitive Information 245
File Transfer 246
Logging 246
Virtual Machine security 247
VNF secure boot 247
VNC Console Access Protection 247
Encrypted VM config data variables 248
Checksum verification for Remote Image Registration 248
Certification Validation for Remote Image Registration 248
VM Isolation and Resource provisioning 248
CPU Isolation 249
Memory Allocation 249
Storage Isolation 250
Interface Isolation 250
Secure Development Lifecycle 251

CHAPTER 11 **Secure Operation in FIPS Mode on NFVIS** 253
Configuring FIPS Mode 253
FIPS Operational Status 256

CHAPTER 12 **NFVIS Clustering Capability for Cisco Cloud Services Platform** 259
Prerequisites for NFVIS Clustering Capability for Cisco CSP 259
Restrictions for NFVIS Clustering Capability on Cisco CSP 260
Information About NFVIS Clustering Capability for Cisco CSP 260
Configure NFVIS Clustering Capability on Cisco CSP 261
Create a Cluster 262
Register and Deploy the VM 262
Migrate the Virtual Machine 263
Virtual Machine Export and Import 264
Delete the Cluster 264
Cluster File Operations 264
Support Commands for NFVIS Clustering Capability 265
Verify NFVIS Clustering Capability 265

NFVIS Clustering Capability Notifications and Syslogs 267

CHAPTER 13 **Platform Specific Configurations** 273

ENCS Switch Configuration 274

ENCS Switch Commands 274

ENCS Switch APIs 274

ENCS Switch Portal Configuration 274

Switch Settings 274

Configuring Spanning Tree 276

Configuring Dot1x 278

Configuring LACP 279

Configuring VLAN 279

Configuring General Settings 280

Configuring Advanced Settings 281

Configuring Spanning Tree per Interface 282

Configure Storm Control 283

Configure Additional WAN Transport Connectivity On ENCS Switch Ports 284

Prerequisites to Configure Additional WAN Transport Connectivity On ENCS Switch Ports 284

Configure Additional WAN Transport Connectivity Using MPLS 284

Configure Additional WAN Transport Connectivity Using Biz Internet 285

Configure Additional WAN Transport Connectivity Using Public Internet 287

Configuration Example of Additional WAN Transport Connectivity to Cisco ISRv or Catalyst 8000V 288

Verify Mapping of LAN SR-IOV with Cisco ISRv or Cisco Catalyst 8000V Using Cisco ISRv or Cisco Catalyst 8000V CLI 288

Configuring vBranch High Availability 289

Prerequisites for vBranch HA 289

SD-Branch HA Design and Topology 289

Isolating LAN and Transit Link Traffic for vBranch HA 292

Enable Port Tracking and Virtual NIC Update 292

Packet Flow for SD-Branch HA 293

Configuration Examples and Usage Description 296

CIMC Secure Overlay Support 301

CIMC TACACS Support 302

LTE PIM Module on Cisco Catalyst 8200 UCPE 303
Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE 322
Information About Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE 322
Benefits of Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE 323
Configure Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE 323
Control Connections via Cellular 323
VM Data Traffic via Cellular 323
Verify Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE 324
External Storage for Cisco ENCS 5400 and Cisco Cloud Services Platforms 325
Support for 40G Dual Port and Quad-Split NICs on Cisco Cloud Services Platforms 327
Prerequisites for 40G Dual Port and Quad-Split NICs on Cisco CSP 327
Restrictions for 40G Dual Port and Quad-Split NICs on Cisco CSP 327
Information About 40G Dual Port and Quad-Split NICs on Cisco CSP 327
Configure 40G Dual Port and Quad-Split NICs on Cisco CSP Using the CLI 328
Configuration Examples for 40G Dual Port and Quad-Split NICs on Cisco CSP 328
Verify PNIC Mode Change on a 40G Dual Port and Quad-Split NICs on Cisco CSP 329
C8300-UCPE Virtual RAID and Supported NIMs Specifications 330
C8300-UCPE Virtual RAID and Supported NIMs Specifications 330
Prerequisites for RAID controller on C8300-UCPE 330
Restrictions for RAID and NIMs on C8300-UCPE 330
Managing Virtual RAID on C8300-UCPE 331
Enable Virtual RAID on the C8300-UCPE 331
Disable Virtual RAID on the C8300-UCPE 332
Rebuild Virtual RAID on the C8300-UCPE Platform 332
Verify Virtual RAID on C8300-UCPE 333

CHAPTER 14 **NFVIS Logging** 335
Configuring System Logs 335

CHAPTER 15 **NFVIS Monitoring** 337
Syslog 337
NETCONF Event Notifications 339
SNMP Support on NFVIS 340
Introduction about SNMP 340

SNMP Operations 340
SNMP Get 340
SNMP Notifications 342
SNMP Versions 342
SNMP MIB Support 343
Configuring SNMP Support 347
System Monitoring 352
Collection of System Monitoring Statistics 353
Host System Monitoring 353
VNF System monitoring 357
ENCS Switch Monitoring 359

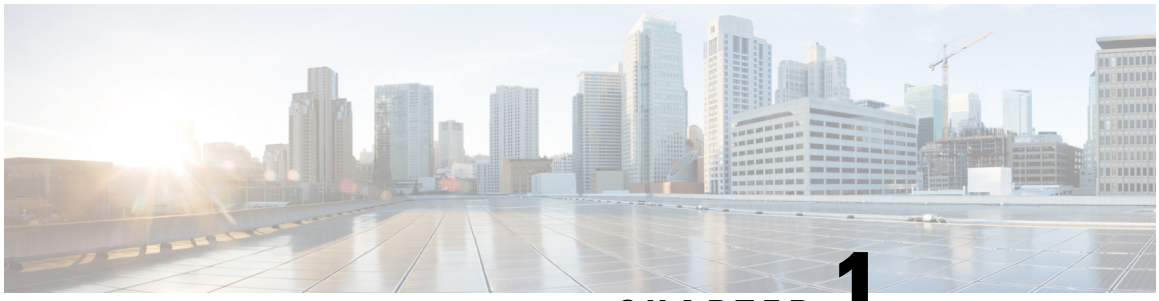
CHAPTER 16 **BIOS Upgrade for C8200 UCPE 361**
Information About BIOS Upgrade for C8200 UCPE 361
Prerequisites for BIOS Upgrade for C8200 UCPE 362
Supported Platforms for BIOS Upgrade 362
Perform BIOS Upgrade for C8200 UCPE 362
Verify BIOS Upgrade for C8200 UCPE 362

CHAPTER 17 **Troubleshoot and Debug Cisco NFVIS 365**
Log and Show Commands 365
SPAN Session or Port Mirroring 367
About SPAN Sessions 367
Configuring SPAN Sessions 367
Configuration Examples for SPAN Session Scenarios 369
Example: SPAN Session Traffic on a Physical Interface 369
Example: SPAN Session Traffic on a LAN SRIOV 370
Example: SPAN Session Traffic on a VLAN 371
Configuring Packet Capture 371

CHAPTER 18 **Appendix 375**
Event Notifications 375
nfvisEvent 375
vmlcEvent 391

Syslog Messages 408
Support for 40G Dual Port and Quad-Split NICs on Cisco Cloud Services Platforms 413
NFVIS Backup and Restore 413
Return Material Authorization (RMA) 414
Granular Role Based Access Control 416
Capability Matrix 416

CHAPTER 19 Glossary 421



CHAPTER 1

New and Changed Information

The following tables summarize the new and changed features and tell you where they are documented.

Table 1: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.15.1

Feature	Description
Cisco ENCS 5400 End of Life (EOL)	Cisco ENCS 5400 has reached End of Life (EOL). The last supported software versions for Cisco ENCS 5400 are Cisco NFVIS Release 4.15.x and Cisco Catalyst SD-WAN Manager Release 20.15.x.

Table 2: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.13.1

Feature	Description
Cisco NFVIS Support on Cisco UCS C-Series M6 Rack Servers	Starting from Cisco NFVIS Release 4.13.1, Cisco NFVIS software is supported on Cisco UCS M6 Rack Servers as a term-based license. Order Cisco NFVIS software smart license along with the Cisco UCS C M6 Rack Server Product purchase.

Table 3: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.12.1

Feature	Description
DHCP Support for Internal Management Network	Cisco NFVIS uses the IP address assigned by a DHCP server for monitoring the VNF deployment.
VM Image Registration Using FTP and SCP Servers	In addition to already existing support to register a VM image using a http or https servers, you can register the VM images using FTP and SCP servers.

Table 4: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.11.1

Feature	Description
Cisco NFVIS ThousandEyes Support	With this feature, you can deploy Cisco ThousandEyes monitoring agent in container form-factor natively on Cisco NFVIS-enabled platforms.
SNMP Cisco-MIB	Use the CISCO MIB 1.3.6.1.4.1.9.1.2376 to display the Cisco NFVIS hostname in SNMP notifications.

Table 5: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.10.1

Feature	Description
NFVIS resource allocation enhancement	The NFVIS system memory allocations are changed for Cisco NFVIS Release 4.10.1.
Configure Additional WAN Transport Connectivity on ENCS Switch Ports	Starting from Cisco NFVIS Release 4.10.1, Cisco ENCS supports terminating WAN circuits on its built-in eight port switch. This is in addition to the two dedicated WAN ports thereby providing you the ability to terminate upto ten WAN circuits on Cisco ENCS devices.
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.
Use **router bgp** command	Starting from Cisco NFVIS Release 4.10.1, the command **route-distribute** is no longer supported. We recommend that you use the **router bgp** command instead.
Use **system settings name-server** command	Starting from Cisco NFVIS Release 4.10.1, the command **dns-server** is no longer supported. We recommend that you use **system settings name-server** command instead.

Table 6: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.9.3

Feature	Description
Change in default system memory for CSP devices	Starting from Cisco NFVIS 4.9.3 release, 8 GB is allocated to Cisco NFVIS in a CSP device with 64 GB memory space.

Feature	Description
A new command to clear cache memory	The **`support flush cache memory`** command is introduced to clear cache memory using the CLI, that clears up some system memory to enable Cisco NFVIS to perform better. Clearing caches using support flush cache command can help resolve issues related to outdated or corrupted cache data. For example, clearing the cache in a web browser can help resolve issues such as slow page load times of Cisco NFVIS portal.

Table 7: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.9.1

Feature	Description
Cisco NFVIS Kubernetes Support	This feature enables you to deploy and operate Kubernetes as a VM using Cisco Enterprise NFVIS. You can download the NFVIS Kubernetes.tar.gz file, upload it to the Cisco Enterprise NFVIS and deploy the Kubernetes VM to leverage Kubernetes related technologies within Cisco Enterprise NFVIS.
VM Image Packaging Tool	The VM image packaging tool provides the UI for you to generate a full VM package, generate a scaffold package, and repackage. The tool works separately from the Cisco NFVIS system.

Table 8: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.8.1

Feature	Description	Where Documented
NFVIS Clustering Capability for Cisco Cloud Services Platform (CSP)	This feature enables you to combine three nodes into a single cluster definition whereby, all the member nodes display configuration information about all the VMs deployed in the cluster. Each member node contributes to a user-specified disk size (same value for all the members) of their total disk size to form a shared and clustered file system. The data is replicated across all the cluster members. This clustering capability is resilient against single node failures. **Note** This capability is a beta feature in this release and should not be used in production because, Cisco may make changes in the subsequent releases without providing any backward compatibility. The beta is provided as-is and without any warranty of any kind.	NFVIS Clustering Capability for Cisco Cloud Services Platform (CSP)
Single IP Monitoring External Gateway	This feature enables you to configure the external gateway monitoring interval, using the **`wan-monitor-interval`** command.	Single IP Monitoring External Gateway
Support for Network VLAN Range Configuration	This feature enables you to configure a VLAN range in trunk mode in the network configuration.	Support for Network VLAN Range Configuration

Table 9: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.7.1

Feature	Description	Where Documented
Granular Role Based Access Control	This feature introduces a new resource group policy that manages the VM and VNF. You can now assign users to a group to control VNF access, during VNF deployment.	Granular Role Based Access Control

Feature	Description	Where Documented
Enforce Setting of BIOS and CIMC Passwords	This feature enforces the user to change the default password for CIMC and BIOS.	Enforce Setting of BIOS and CIMC Passwords
Support for 40G Dual Port and Quad-Split NICs in Cisco Cloud Services Platforms	Starting from this release, the 40G network interface card (NIC) supports dual port modes on Cisco Cloud Services Platform (CSP).	Support for 40G Dual Port and Quad-Split NICs in Cisco Cloud Services Platforms

Table 10: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.6.1

Feature	Description	Where Documented
Local Authentication for a Specific Group of Users	This feature allows you to create a group with specific users, who can perform only the local authentication; and don't have to authenticate externally through TACACS.	Local Authentication for a Specific Group of Users
Support for External Storage for Cisco Cloud Services Platforms	External disks are supported for Cisco Cloud Services Platforms (CSP).	Support for External Storage for Cisco Cloud Services Platforms
Support for Replacing HTTP Basic Authentication	This feature enhances NFVIS local portal capabilities, to replace HTTP basic authentication to prevent accidental leakage of credentials.	–
Support for NFVIS Container Life Cycle Management	This features provides support for container lifecycle management.	Support for NFVIS Container Life Cycle Management

Table 11: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.5.1

Feature	Description	Where Documented
ENCS Switch Monitoring	This feature allows you to calculate the data rate for ENCS switch ports based on the data collected from the ENCS switch.	ENCS Switch Monitoring, on page 359
Authentication Cache for External Authentication Server	This feature supports TACACS authentication through OTP on NFVIS portal	Authentication Cache for External Authentication Server, on page 231

Feature	Description	Where Documented
BGP Route Annoucement over MPLS or IPSec	This feature allows you to configure NFVIS to announce routes through BGP over MPLS or in conjunction with secure overlay to announce routes over IPSec tunnel.	BGP Route Annoucement over MPLS or IPSec

Table 12: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.4.1

Feature	Description	Where Documented
Cisco Catalyst 8200 Series Edge Universal CPE Support	The Cisco Catalyst 8200 Series is the next generation of Cisco Enterprise Network Compute System platform that combines routing, switching and application hosting into a compact one rack unit device. 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.	Supported Hardware Platforms On NFVIS Cisco Catalyst 8200 Series Edge uCPE Hardware Installation Guide
SNMP VM Monitoring MIB	SNMP VM monitoring MIBs support added.	SNMP MIB Support, on page 343
Live Export Support	Supports VM backup process without shutting down the VM.	VM Action APIs
VM Export with Selective Disk	You can exclude certain disks or volumes from a VM export.	VM Export with Selective Disk, on page 142
BGP Support on NFVIS	This feature allows NFVIS system to learn routes announced from the remote BGP neighbor and apply the learnt routes to NFVIS system. This feature also allows you to announce or withdraw NFVIS local routes to the remote BGP neighbor.	BGP Support on NFVIS
DNS Name Server Configuration Enhancement	NFVIS allows configuring upto three nameservers which can be used by DNS resolvers in the order specified by the user. New command introduced to configure DNS nameservers: **system settings name-server**.	Configure DNS Name Servers, on page 87

Feature	Description	Where Documented
CIMC Secure Overlay Support	This is a disaster recovery feature to recover an unresponsive NFVIS. CIMC periodically monitors NFVIS health and when NFVIS health check fails, CIMC establishes a secure tunnel to the remote. At remote site, you can login to CIMC and do the necessary troubleshooting to recover NFVIS.	CIMC Secure Overlay Support, on page 301

Table 13: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.2.1

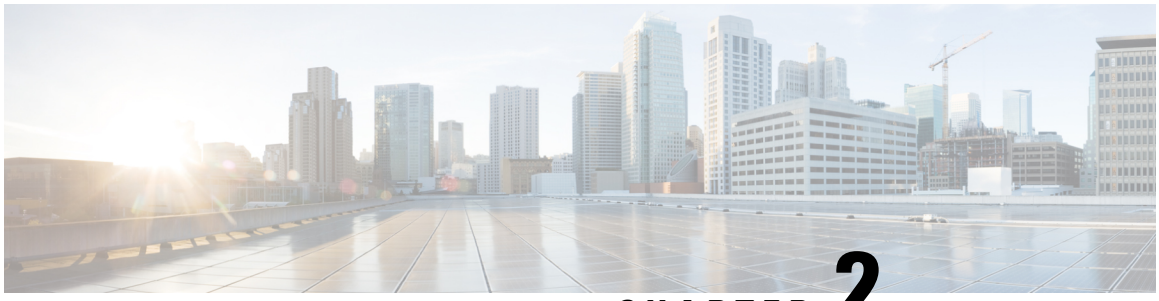
Feature	Description	Where Documented
Enhancements to backup and restore of configurations	New commands are introduced to view the overall status of backup and restore process. Enhancements to backup file location and factory default options are introduced. Information on how to troubleshoot failure to restore NFVIS configurations is added.	Backup and Restore NFVIS and VM Configurations, on page 75
HugePage memory and CPU allocation	The system memory allocations are enhanced and all memory apart from the amount reserved for system is converted to HugePage memory.	Host System Requirements, on page 12
Command for PnP certificates	A certificate can be used as a PnP root certificate through Command Line Interface (CLI).	Certificate for Static PnP discovery, on page 66
Secure Operation in FIPS Mode on NFVIS	The Federal Information Processing Standards (FIPS) Publication 140-2 are publicly announced standards developed by the United States federal government for use in computer systems by non-military government agencies and government contractors.	Secure Operation in FIPS Mode on NFVIS
BIOS and CIMC password	New password restrictions and security measures are added for CIMC and BIOS.	BIOS and CIMC Password, on page 24
CIMC TACACS Support	CIMC TACACS configuration support enabled on NFVIS.	CIMC TACACS Support, on page 302

Table 14: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 4.1.1

Feature	Description	Where Documented
Storm control support	Storm control is used to monitor incoming traffic levels and limit excessive flow of packets on any user facing switch port that could cause a traffic storm.	Configure Storm Control, on page 283
hot-add and hot-delete support on DPDK interface	Support interface hot add and hot delete operations on DPDK networks.	Update VM Image, on page 117
VM backup and restore perserving mac and UID	Specifying the unique MAC ID flag during VM import will ensure that the imported VM is not deployed with the same UID and interface MAC addresses.	Import and Export NFVIS VM, on page 159
Support placing VM's extra storage volumes to external datastore	This feature supports placing VNF volumes on external datastore storage locations	VNF Volumes, on page 142
PNIC promiscuous mode support	Enabling promiscuous mode on an interface can be used to monitor all incoming packets on the interface.	Promiscuous mode, on page 57

Table 15: New and Changed Features for Cisco Enterprise Network Function Virtualization Infrastructure Software, Release 3.12.3

Feature	Description	Where Documented
Secure Tunnel enhacements for PKI and EAP	EAP and PKI authentication supported on secure overlay.	Secure Overlay, on page 197
APC UPS support	This feature provides support for monitoring battery status for an APC UPS connected to the ENCS box through a USB cable.	APC UPS Support and Monitoring, on page 83
SNMP support for CPU usage	Supported SNMP MIBs for CPU usage	SNMP MIB Support, on page 343
AAA auth-order	In this feature the supported aaa authentication order is local authentication followed by TACACS+.	Default Authentication Order, on page 42



CHAPTER 2

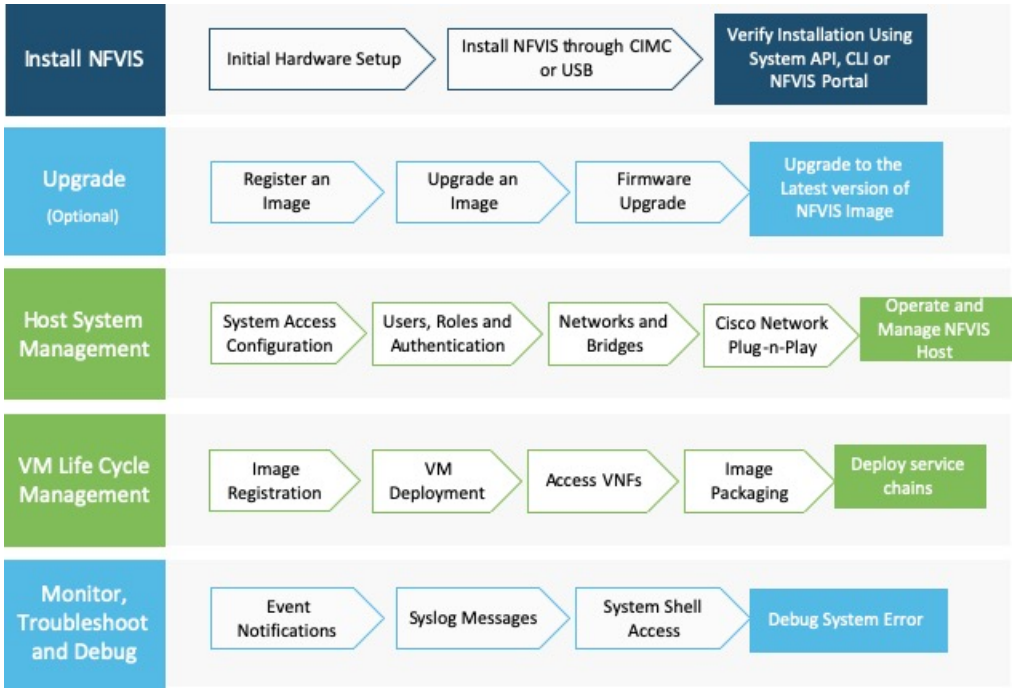
About NFVIS Documentation

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

This chapter describes how documentation about Cisco NFVIS is structured.

NFVIS Workflow

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

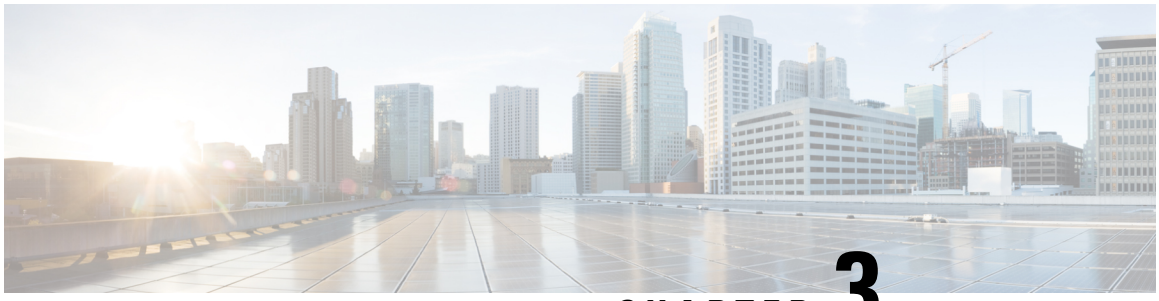


NFVIS documentation are structured according to the stages an enterprise or service provider would go through when they decide to setup NFVIS:

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

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

- The Configuration Guide provides detailed information once you have completed the basic installation and set up. This guide covers managing the host system, registering and managing VNFs, security cosiderations for your network, troubleshooting issues with NFVIS and so on.
  - Host System Management: Provides detailed information about operations and management of NFVIS host.
  - #unique_24: Provides information on the entire process of registering, deploying, updating, monitoring VMs, and getting them service chained as per your requirements.
  - Troubleshoot and Debug Cisco NFVIS, on page 365: Information provided here helps troubleshoot and debug system errors.



CHAPTER 3

Host System Management

Table 16: Feature History

Feature Name	Release Information	Feature Description
NFVIS Portal Language Preferences	Cisco NFVIS Release 4.18.1	The NFVIS portal now supports both English and Japanese languages. Administrators can set the preferred language for new user accounts, ensuring the portal displays in that language upon first login. Existing users can also change their interface language via user settings.

- System Access Configuration, on page 12
- Users, Roles, and Authentication, on page 27
- Networking, on page 43
- **Cisco Network Plug-n-Play Support** , on page 60
- DPDK Support on NFVIS, on page 68
- Storage Access, on page 71
- Host System Operations, on page 71
- Backup and Restore NFVIS and VM Configurations, on page 75
- APC UPS Support and Monitoring, on page 83
- Resetting to Factory Default, on page 84
- Configure Banner, Message of the day and System Time, on page 85
- Configure DNS Name Servers, on page 87
- Configuring IP Host, on page 88

System Access Configuration

Host System Requirements

Table 17: Feature History

Feature Name	Release Information	Description
HugePage memory and CPU allocation	NFVIS Release 4.2.1	The system memory allocations are enhanced and all memory apart from the amount reserved for system is converted to HugePage memory. The VMs are deployed using hugepage memory even when DPDK is enabled on the system.
NFVIS resource allocation enhancement	NFVIS Release 4.10.1	The system memory allocations are updated.

The following resources are required for a standalone Cisco Enterprise NFVIS:

Table 18: CPU Allocation

Total Cores	Before NFVIS 3.12.x Release	NFVIS 3.12.x till NFVIS 4.9.x	NFVIS 4.10.x and later Releases
12 or less	1	1 + (1 core per socket applicable to DPDK systems)	1 + (1 core per socket applicable to DPDK systems)
Between 12 and 16 (including 16)	2	1 + (1 core per socket applicable to DPDK systems)	1 + (1 core per socket applicable to DPDK systems) + (1 if Multi-NUMA node system*)
More than 16	4	2 + (1 core per socket applicable to DPDK systems)	2 + (1 core per socket applicable to DPDK systems) + (1 if Multi-NUMA node system*)
* Indicates that Multi-NUMA node systems require an additional CPU core system reserved. This additional core is helpful in processing the cross NUMA nodes, indirectly improving the performance of Cisco NFVIS functions on the system cores.			

Note

- If hyper-threading is enabled on the device, each core reflects two logical CPUs.

Table 19: Memory Allocation

Reserved System Memory	Up to 16 GB	Up to 32 GB	Up to 64 GB	Up to 128 GB	Greater than 128 GB	Greater than 256 GB
Reserved for NFVIS 3.12.x and earlier releases	3 GB	3 GB	4 GB	4 GB	4 GB	NA
Reserved for NFVIS 3.12.x and 4.1.x releases	3 GB	3 GB	4 GB	8 GB	8 GB	NA
Reserved for NFVIS 4.2.1 release	3 GB	3 GB	4 GB	8 GB	16 GB	NA
Reserved for NFVIS 4.2.2 till NFVIS 4.8.1	3 GB	4 GB	4 GB	8 GB	16 GB	NA
Reserved for NFVIS 4.9.1	3 GB	4 GB	4 GB	8 GB/ 10 GB*	16 GB/ 20 GB*	NA
Reserved for NFVIS 4.10.1 till NFVIS 4.11.1	5 GB/ 6 GB*	5 GB/ 6 GB*	5 GB/ 6 GB*	8 GB/ 10 GB*	16 GB/ 20 GB*	NA
Reserved for NFVIS 4.12.1 till NFVIS 4.14.1	5 GB/ 6 GB*	5 GB/ 6 GB*	5 GB/ 6 GB*	11 GB/ 13 GB*	16 GB/ 20 GB*	NA
Reserved for NFVIS 4.15.1 or later releases	5 GB/ 6 GB*	5 GB/ 6 GB*	5 GB/ 6 GB*	11 GB/ 13 GB*	16 GB/ 20 GB*	NA
	For UCSC-M6 : 7 GB/ 8 GB*	For UCSC-M6 : 11 GB/ 12 GB*	For UCSC-M6 : 11 GB/ 12 GB*	For UCSC-M6 : 11 GB/ 12 GB*	For UCSC-M6 : 11 GB/ 13 GB*	For UCSC-M6 : 16 GB/ 20 GB*
		For C8300-UCPE-1N20 : 8 GB/ 9 GB*	For C8300-UCPE-1N20 : 13 GB/ 14 GB*	For C8300-UCPE-1N20 : 13 GB/ 14 GB*	For C8300-UCPE-1N20 : 13 GB/ 15 GB*	For C8300-UCPE-1N20 : 16 GB/ 20 GB*
***** Indicates the memory allocation is applicable only for Multi-NUMA node systems. In case of single node systems, the memory allocation values without ***** is applicable.						

Note

- ***** indicates that the memory allocation is applicable only for Multi-NUMA node systems. In case of other systems, the memory allocation remains the same that is reserved for NFVIS Release 4.2.2 till NFVIS 4.8.1
- When you upgrade Cisco NFVIS from Cisco NFVIS Release 4.9.x to any of the later releases, some VMs may not function as there is not enough system memory allocated. When the memory utilization of the VMs used in Cisco NFVIS is equal to or closer to the total amount of memory allocated for the VMs, the VMs fail to function. The system allocates the memory for Cisco NFVIS and the remaining memory is converted into hugepage memory. From the hugepage memory, Cisco NFVIS allocated memory to DPDK and VMs when the memory is exhausted.

Total System Memory	Additional memory required for DPDK support per NUMA node
Upto 63 GB	1
64 GB - 127 GB	2
128 GB - 256 GB	4

Starting from NFVIS 4.2 release, all memory apart from the amount reserved for system is converted to HugePage memory. The system memory is allocated on socket 0 on system with multiple CPU sockets.

Note When you use Cisco Catalyst Edge uCPE 8300 for high throughput requirements, we recommend that you use NVME based storages (M.2 NVME or U.2 NVME) or E1.S based.

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

System Setting Hostname

You must adhere to the following rules for hostname on NFVIS:

- Must contain minimum length of 2 and maximum length of 255.
- Must begin with a letter or digit and can contain alphabets, numbers and hyphen.
- Must not be deleted.
- The hostname range is from 1 to 58. The hostname range must contain a letter or a digit, it may contain alphabets, numbers, and hyphens.

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.

Restrictions for Dual WAN Support

- The DHCP toggle behavior is not supported during the upgrade flow. It is only triggered during fresh installation of NFVIS or after a factory default reset.
- Active/standby or redundant WAN bridges are not supported. NFVIS does not detect connectivity failure from one WAN bridge and switchover to another WAN bridge. In case connectivity fails on the WAN bridge with DHCP configurations, connectivity through the other WAN bridge is established only if static IP is applied to the second WAN bridge and static routing is configured for connectivity through that bridge.
- IPv6 is not supported for dual WAN toggle.
- If wan2-br is DHCP enabled WAN bridge, you must remove DHCP from wan2-br to apply default gateway from static IP configurations.

Dual WAN Bridge and DHCP Toggle



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

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

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 60).
- 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 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 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

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 20: 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 newpassword • hostaction change-bios-password newpassword

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.

PXE Boot Mode

PXE (Preboot Execution Environment) boot mode is a new configuration option in the BIOS **Advanced** option list in CIMC, which can be configured like any other BIOS configuration option in the list. PXE boot mode allows you to configure PXE boot for legacy mode, UEFI mode, or disable it when not in use. Starting from NFVIS 4.5, BIOS 2.13 and CIMC 3.2.12.1, PXE boot is disabled by default.

It is recommended that you disable PXE boot mode when not using PXE, in order to gain boot time savings.

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

Users, Roles, and Authentication

Local User Account Management

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

Table 21: Supported User Roles and Privileges

User Role	Privilege
Administrators	Owns everything, can perform all tasks including changing user roles, but cannot delete basic infrastructure. An admin's role can't be changed
Operators	Start, stop, and delete a VM. Clear logs and view all information
Auditors	Read-only permission and can't perform any tasks

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 (*), or question mark (?).
- Must contain an uppercase character and a lowercase character.
- Must not be the same as last five passwords.

Creating Users and Assigning Roles

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



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

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

To create a user:

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

To delete a user:

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



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

User Management APIs and Commands

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

Configuring Minimum Length for Passwords

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

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

Minimum Password Length APIs and Commands

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

Configuring Password Lifetime

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

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



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

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

Password Lifetime APIs and Commands

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

Deactivating Inactive User Accounts

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

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

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

Deactivate Inactive User Accounts APIs and Commands

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

Activating an Inactive User Account

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

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

Activate Inactive User Account APIs and Commands

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

NFVIS Password Recovery

1. Load the NFVIS ISO image, using the CIMC KVM console.
2. Select Troubleshooting from the Boot Selection menu.
3. Select Rescue a NFVIS Password.

4. Select Continue.
5. Press Return to get a shell.
6. Run the **chroot /mnt/sysimage** command.
7. Run the **./nfvis_password_reset** command to reset the password to admin.
8. Confirm the change in password and enter Exit twice.

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

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



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

User and Role Management in the NFVIS Portal

The Cisco Enterprise NFVIS web portal provides a graphical interface for managing user accounts and roles, offering an intuitive alternative to command-line operations. Administrators can perform various tasks to control access and privileges within the system, including easily creating, modifying, and deleting user accounts, and assigning them specific roles to define their access levels.

Create New Users

To create a new user account through the NFVIS portal:

1. Log in to the NFVIS portal as an administrator.
2. Navigate to **Configuration** > **Host** > **Security** > **Users and Roles**.
3. From the **Users and Roles** page, click the **+** icon to create a new user.
4. Enter the following details:

Field	Description
Name	Enter the unique login name for the new user account..

Field	Description
Role	Select the predefined role that defines the user's permissions and access level within the NFVIS system. Options typically include: • Administrator: Full access to configure, update, or delete resources, and manage user roles. • Operator: Can view, start, stop, and delete Virtual Machines (VMs). • Auditor: Read-only access to system information.
Password	Enter the password for the new user account. This must comply with the configured strong password policies (e.g., minimum length, complexity requirements including uppercase, lowercase, numbers, and special characters).
Confirm Password	Re-enter the password exactly as entered in the **Password** field to confirm accuracy and prevent typing errors.
Group	Choose a user group to assign this user to. User groups are used for granular role-based access control (RBAC) and can define specific access policies for sets of users or resources.
Preferred Language	Choose the default language for a user's NFVIS portal interface: • English • Japanese When this user logs in, the NFVIS portal will be displayed in the language chosen here.

5. Click **Submit** to create the new user account with the specified details and save the configuration.

   Click **Cancel** to discard any entered information and return to the previous screen without creating the user.

   Click **Reset** to clear all fields on the form, allowing you to re-enter information.

Modify Users

To modify a user account through the NFVIS portal:

1. Log in to the NFVIS portal as an administrator.
2. Navigate to **Configuration** > **Host** > **Security** > **Users and Roles**.
3. From the **Users and Roles** page, click the edit icon to modify an existing user account.

When modifying a user account, you can update details such as their Role, password, Group, and Preferred Language..

4. Click **Submit** to apply the changes and save the updated configuration.

   Click **Cancel** to discard any changes and return to the previous screen without modifying the user.

   Click **Reset** to clear all fields on the form, allowing you to re-enter information.

Delete Users

To delete a user account through the NFVIS portal:

1. Log in to the NFVIS portal as an administrator.
2. Navigate to **Configuration** > **Host** > **Security** > **Users and Roles**.
3. From the **Users and Roles** page, identify the user account you want to delete and click the Delete icon associated with it.

   A confirmation message is displayed to confirm the deletion of the user account.
4. In the confirmation dialog, use the following options:
   - Delete: Click this button to proceed with the permanent deletion of the selected user account(s).
   - Cancel: Click this button to abort the deletion process and return to the "Users and Roles" page without removing the user(s).

Change Language Preferences in the NFVIS Portal

Starting from NFVIS 4.18.1, Cisco NFVIS portal now supports both English and Japanese languages, providing users with the flexibility to interact with the interface in their preferred language.

- Users can adjust their language preferences through two ways methods within the NFVIS portal:
  - From the current user session, using the **Settings** icon.
  - From the Users and Roles page. For more information on setting the default language for user accounts, see Create New Users, on page 31.

Change Language Preferences Using the Settings Icon

To change the language for your current portal session:

1. Log in to the NFVIS portal
2. Click the user profile icon.
3. From the dropdown menu, choose **Language Preferences**
4. Choose your desired language (English or Japanese) from the available options.

   Once a language is chosen, the portal interface will immediately update to the chosen language, and a confirmation notification will appear. This change applies only to your current session.

User Groups

Granular Role-Based Access Control

Table 22: Feature History

Feature Name	Release Information	Description
Granular Role-Based Access Control	NFVIS 4.7.1	This feature introduces a new resource group policy that allows the system administrator to define a set of user groups. You can assign users to these defined group to control VNF access, during a VNF deployment.

Restrictions for Granular Role-Based Access Control

- A group can only be associated with one policy, either the `resource-access-control` policy or the `local-authentication-only` policy.
- One user can be assigned to one group only.
- A VM can only belong to one group.

Information About Granular Role-Based Access Control

The Granular Role-Based Access Control (Granular RBAC) feature allows you to restrict VM management to a particular set of users. The system administrator can define a set of resource groups, and assign VNFs and system resources such as VMs, disk files, and system level configurations to these defined groups. When you create a user, you can assign that user to one of the resource groups, and this enables the user to access the associated VNFs. Following is the detailed descriptions of roles, users, and groups.

Roles

The three roles defined by the system are:

- Administrator: An administrator user has complete access to configure, update or delete a resource.
- Operator: An operator user can only view and operate a resource.
- Auditor: An auditor user can only view a resource and cannot perform any action on it.



Note All three roles have a read-access to all host level configurations, VMs, and images.

Users

- Users are created with a role definition. A user's role is defined during the user creation, and it is consistent across all groups.
- All users have a read access to NFVIS configurations, filesystems, logs, VMs and images.

- A user can be a member of only one group.
- admin user:
  - The admin user is a special user that cannot be deleted or modified. The admin user permanently has the administrator role in the default global group.
  - The admin user functions as a member of every group, and can execute administrative privileges for every group.
  - The admin user cannot be assigned to a specific user group.
- The roles of remote users such as TACACS and RADIUS, is mapped to NFVIS roles based on the privilege level.

Groups

- The global group is a special group assigned to users who are not members of any other group. A user in the global group can access all resources on the system, at the privilege level.
- Resources can be assigned to the global group or a specific group.
- When creating a group, the `resource-access-control` policy should be enabled, to have resource restrictions.
- All members of a group have access to the resources that are assigned to that group, the privileges of which are defined by the user's role.
- A resource can belong to one specific group, and all users that are members of that group containing a resource have access to that resource.

For more details on Granular RBAC feature capabilities, see Appendix.

Create Groups and Assign Local Users to the Groups

The following example shows how to create a group with or without a policy:

```
nfvis(config)# aaa groups group rac_group policy resource-access-control
nfvis(config-policy-resource-access-control)# commit
Commit complete.
```

The following example shows how to create a local user and assign them to a group:

```
nfvis# rbac authentication users create-user name local_admin_3 password Cisco123\# role
administrators group rac_group
```

The following example shows how to view a list of RBAC users with role and group information:

```
nfvis# show running-config rbac authentication users
rbac authentication users user admin
 role administrators
!
rbac authentication users user local_admin_1
 role administrators
!
rbac authentication users user local_admin_2
 role administrators
!
rbac authentication users user local_admin_3
```

```
 role administrators
 groups group rac_group
 !
!
rbac authentication users user local_oper_1
 role operators
!
rbac authentication users user local_test_1
 role operators
!
```

Assign Remote Users to the Group

NFVIS depends on a remote server for user authentication and authorization. Hence, NFVIS uses a different way to assign remote users to a resource control group. When remote users login to NFVIS, they can only manage the deployment that belongs to its own resource control group, based on their defined role in the remote server. Any remote user that is not mentioned in the resource group, is treated as a global group user and that user can operate the system and manage the deployments based on their defined role.

The following example shows how to assign remote users to the group:

```
nfvis(config)# aaa groups group tac_group user remote_admin1
nfvis(config-user-remote_admin1)# user remote_admin2
nfvis(config-user-remote_admin2)# user remote_operator3
nfvis(config-user-remote_operator3)# policy resource-access-control
nfvis(config-policy-resource-access-control)# commit
Commit complete.
nfvis(config-policy-resource-access-control)# end
nfvis# show running-config aaa groups group tac_group
aaa groups group tac_group
policy resource-access-control
!
user remote_admin1
!
user remote_admin2
!
user remote_operator3
!
!
nfvis#
nfvis# show rbac authentication users
NAME
-------
admin
```

Note In the above example, `remote_admin1`, `remote_admin2`, and `remote_operator3` are TACACS+/RADIUS users.

Local Authentication for a Specific Group of Users

Table 23: Feature History

Feature Name	Release Information	Description
Local Authentication for a Specific Group of Users	NFVIS 4.6.1	This feature enables you to create a group with specific users, who can perform only the local authentication; and don't have to authenticate externally through TACACS.

Restrictions for Local Authentication for a Specific Group of Users

- One user can be assigned to a maximum of one group.
- If a local user assigned to the local authentication group has the same user name as a remote (TACACS+/RADIUS) user, then only the local user's credentials are taken into consideration. The remote user's credentials are considered even if the local user's authentication fails.

Information about Local Authentication for a Specific Group of Users

Typically, users who need authentication go through the default authentication order. The default order involves external authentication through TACACS as the first step, and local authentication as the second step. The local authentication for a specific group of users feature enables you to create a group and add specific users to it, allowing these users to bypass the default authentication order. The users in this group skip the external TACACS authentication and only go through local authentication. For this group to function as expected, assign the 'local-authentication-only' policy to the group.

Create Groups and Assign Users to the Groups

The following example shows how to create a group:

```
nfvis#(config) aaa groups group [ group-name ] policy local-authentication-only
```

The following example shows how to assign a user:

```
nfvis# rbac authentication users user <username> assign-group [ group-name ]
```

The following example shows how to remove a user:

```
nfvis# rbac authentication users user <username> remove-group [ group-name ]
```

The following example shows how to assign a user to a group while creating the user:

```
nfvis# rbac authentication users create-user name <username> password <password> role <role>
 group [ group-name ]
```

RADIUS Support

About RADIUS

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

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

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



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 RADIUS

To configure RADIUS support:

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

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

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

Verifying the RADIUS configuration

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

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

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

RADIUS Support APIs and Commands

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

TACACS+ Support

About TACACS+

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

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

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

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

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

TACACS Operation

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

1. When the user tries to log in, NFVIS sends user credential to TACACS+ server.
2. NFVIS will eventually receive one of the following responses from the TACACS+ server:
   a. ACCEPT—The user is authenticated and service can begin. If NFVIS is configured to require authorization, authorization begins at this time.
   b. REJECT—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ server.
   c. ERROR—An error occurred at some time during authentication with the server or in the network connection between the server and NFVIS. If an ERROR response is received, NFVIS typically tries to use an alternative method for authenticating the user.
   d. CONTINUE—The user is prompted for additional authentication information.

   After authentication, NFVIS will send authorization request to TACACS+ server.
3. Based on authorization result, NFVIS will assign user's role.

Configuring a TACACS+ Server

To configure TACACS+:

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

key 0
admin-priv
14

oper-priv
9

commit
```

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

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

To configure encrypted TACACS+ key:

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

oper-priv
9

commit
```

Verifying the TACAC+ configuration

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

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

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

TACACS+ APIs and Commands

TACACS+ APIs	TACACS+ Commands and Descriptions
• /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 IP address of the TACACS host server • key Configures a preshared key to authenticate comms between Security client and server. • admin-priv Minimum privilege level for administrator • oper-priv Minimum privilege for operator • encrypted-shared-secret or shared-secret Preshared secret to authenticate comms between Security client and server.

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 337 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)#
```

Networking

-

Bridges

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

NFVIS is installed with LAN and WAN bridges by default. A service bridge can also be created. On all NFVIS systems, lan-br and wan-br are generated by default and populated with the appropriate ports for that system. On ENCS 5000 series platforms, wan2-br is also generated by default for the dual WAN initialization. For more information, see Dual WAN Support, on page 15. Except on ENCS 5000 Series platforms, the default LAN bridge is configured with a static IP address 192.168.1.1 and the WAN bridges uses DHCP for initial NFVIS connectivity.

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



Note On UCSC-M6 platforms, a default wan2-br bridge is not available.

Using IPv4

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

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

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

Using IPv6

IPv6 can be configured in static, DHCP stateful, and Stateless Auto configuration (SLAAC) modes. By default, DHCP IPv6 stateful is configured on the WAN interface. If DHCP stateful is not enabled on the network, the router advertisement (RA) flag decides which state the network stays in. If the RA shows 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 a default gateway. Stateless DHCP is enabled in the SLAAC mode. If the server has DNS and domain configured, then SLAAC also provides those values through stateless DHCP.

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

Creating Bridges

To configure a new bridge:

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

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

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

Configuring Bridge Port

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

To configure a port on a bridge:

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

To configure a port channel on a bridge:

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

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

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

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

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

Configuring Bridge IP Connectivity

Configuring DHCP on Bridge

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

To configure DHCP on a bridge:

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

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

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

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

Configuring Static IP on a Bridge

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

To configure an IPv4 address on a bridge:

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

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

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

Configuring IPv6 DHCP on a Bridge

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

To configure IPv6 DHCP on a bridge:

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

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

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

Configuring IPv6 SLAAC on a Bridge

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

To configure IPv6 SLAAC on a bridge:

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

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

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

Configuring Static IPv6 Address on a Bridge

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

To configure an IPv6 address on a bridge:

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

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

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

Configuring VLAN on a Bridge

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

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

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



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

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

To configure a VLAN:

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

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

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

Configuring MAC Aging Time on Bridge

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

To configure MAC aging time on a bridge:

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

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

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

Bridge APIs and Commands

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

Physical Network Interface Cards

Configuring LLDP

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

LLDP information is sent by devices from each of 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.

Note Starting with Cisco Enterprise NFVIS Release 4.15.5 on Cisco Catalyst 8300 Series Edge uCPE hardware platforms, PNIC auto-negotiation always remains enabled, even when speed and duplex are explicitly configured. Auto-negotiation is always active on the PNIC; therefore, when setting a specific speed (for example, 100 Mbps), configure the desired speed in NFVIS and ensure that the network device connected to the PNIC also has auto-negotiation enabled. This allows both ends to negotiate correctly and ensures the link is established at the intended speed (such as 100 Mbps) for optimal performance.

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

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.

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

Promiscuous mode

Starting from NFVIS 4.1.1 release, NFVIS allows enabling promiscuous mode on interfaces. Enabling promiscuous mode on an interface can be used to monitor all incoming packets on the interface.

To enable promiscuous mode:

```
nfvis# config terminal
nfvis(config)# pnic GE0-0 promiscuous enabled
nfvis(config-pnic-GE0-0)# commit
```

Use the **show pnic GE0-0 operational-promiscuous** command to verify if promiscuous mode has been enabled.

Note When an interface is connected to a bridge, NFVIS enables promiscuous mode on the interface.

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.

System Routes

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

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

Configuring System Routes

The following example shows how to configure additional static routes:

```
configure terminal
system routes route 172.25.222.024 gateway 172.25.221.1
system routes route 172.25.223.024 dev wan-br
commit
```

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

```
nfvis# show system routes

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

System Routes APIs and Commands

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

Troubleshooting

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

```
nfvis# show system routes

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

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

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

The example above indicates that the failure is caused because the network is unreachable. To resolve this issue you can either reconfigure the route with a reachable gateway or identify network connectivity issue.

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

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

Cisco Network Plug-n-Play Support

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

The Cisco Network Plug and Play (Cisco Network PnP) solution provides a simple, secure, unified, and integrated offering for enterprise network customers to ease new branch or campus device rollouts, or for provisioning updates to an existing network. The solution provides a unified approach to provision enterprise networks comprising Cisco routers, switches, and wireless devices with a near zero touch deployment experience. This solution uses Cisco Application Policy Infrastructure Controller Enterprise Module (APIC-EM) to centrally manage remote device deployments.

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

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

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

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

Certificate for Static PnP discovery

Table 24: Feature History

Feature Name	Release Information	Description
Command for PnP certificates	NFVIS 4.2.1	A certificate can be used as a PnP root certificate through Command Line Interface (CLI).

Starting from NFVIS 4.2.1 release, a certificate can be used as a PnP root certificate through Command Line Interface (CLI). The following command is used to upload a certificate:

```
system certificate input filepath <filepath> pem-data <certificate contents>
```

- The file containing the certificate information is created inside /data/intdatastore/uploads directory.
- The certificate should be in PEM encoding. Any invalid content or format is rejected with an error message.
- Multiple certificates can also be added to form a certificate chain and they should be separated by a new line.
- The certificate content must have:
  - Maximum size of base64 content in each certificate limited to 6144 bytes,
  - Maximum number of certificates allowed in a certificate chain input as 10.
- For both single certificate and a chain of certificates, the input should end with a new line.
- If a file with the same name as the certificate file name already exists inside /data/intdatastore/uploads directory, the user gets an appropriate error message.

The following are examples to show how to upload certificates:

```
nfvis# system certificate input filepath intdatastore:uploads/apic_em_oneline_02.pem pem-data
 "-----BEGIN
[redacted]
 CERTIFICATE-----\n-----BEGIN
[redacted]
 CERTIFICATE-----\n"
nfvis#


nfvis# system certificate input filepath
```

```
Value for 'filepath' (<string>): intdatastore:uploads/test_cert.pem
Value for 'pem-data' (<string>):
[Multiline mode, exit with ctrl-D.]
> -----BEGIN CERTIFICATE-----
> MIIDvzCCAqegAwIBAgIUKBZWJ4U2c1gBYJFBFWNlNQSVd2MwDQYJKoZIhvcNAQEL
> BQAwgYgxLTArBgNVBAMTJDEyYjQ0ZDhmLWRhZWItNGIxNi1iYWZhLWRiYTQxZGRk
> MWY3MzELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNhbGlmb3JuaWExEDAOBgNVBAcT
> B1Nhbkpvc2UxEzARBgNVBAsTCkFQSUNFTS1TRE4xDjAMBgNVBAoTBUNpc2NvMB4X
> DTE4MDcyMTIwNDMyN1oXDTIzMDcyMDIwNDMyN1owXjEVMBMGA1UEAxMMMTcyLjI1
> LjIxNy44MRMwEQYDVQQIEwpDYWxpZm9ybmlhMQswCQYDVQQGEwJVUzEOMAwGA1UE
> ChMFQ2lzY28xEzARBgNVBAsTCkFQSUNFTS1TRE4wggEiMA0GCSqGSIb3DQEBAQUA
> A4IBDwAwggEKAoIBAQDD616jSBX+DfxI5kGT3JCmcIgOHYDDhJJd1L8qhxDAOM7P
> qLjLEdUCOTap8Lu/dKpVClN+hE0LNr9HyQSZ7Mn8+UrM7wQqTKEA6p3dw323wWbp
> ia/XSgByznj4JEY3xW8/trKGkoxfZ7D2JA/cLTf7hK3v44Nev4ONdKtnwNXV32ms
> E7Mpx/Wb/110hYn37DJ9sN5+o34F1KkF1quaXPOgmQA8PMH2Y1LMOh/Z3g3afali
> uOsbPpeGxawEWgfc5pK5EJbziWy1JUSh/c2DaqrfvnKnlprM/HfjZLgzit9FnLyL
> 0lxW54U77AMDRcNALtSq0G3YDX12pdb+ateFWznFAgMBAAGjSjBIMAkGA1UdEwQC
> MAAwCwYDVR0PBAQDAgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjAP
> BgNVHREECDAGhwSsGdkIMA0GCSqGSIb3DQEBCwUAA4IBAQAiD3XiYgUTFK0MuxuB
> DYanVIuAPBImqULdidh3uDlnoTUlzDFU+feYug+XHjAPk7rczZ1Yoc5Hvo90/gQU
> WQUd9TftJygftRn4vG4MMb4fjDPqoh7rxj0P51NRJIA64ro6/gvbQn+T70a62/Wt
> sDGIdXhp8MUTzA/R99vxYUS9YfTMFVAEv5KAW9ZRwC+CTPT+YeN0B+iNO3cXmZ3A
> C0YDXgN27yfF/+1Lna1aOMCTXdTFzNCplOIhhWKulM74Re9QN0Lwvkt9PDim3rX+
> GaDvd4nINMcIXRXy40ieQOOv0W9xnYnvgFD3xjhKQhzDEXX1BStXVJwaqAwWlDlY
> 8EdE
> -----END CERTIFICATE-----
> -----BEGIN CERTIFICATE-----
> MIIDrzCCApegAwIBAgIGCMgojJNOMA0GCSqGSIb3DQEBCwUAMIGIMS0wKwYDVQQD
> EyQxMmI0NGQ4Zi1kYWViLTRiMTYtYmFmYS1kYmE0MWRkZDFmNzMxCzAJBgNVBAYT
> AlVTMRMwEQYDVQQIEwpDYWxpZm9ybmlhMRAwDgYDVQQHEwdTYW5Kb3NlMRMwEQYD
> VQQLEwpBUElDRU0tU0ROMQ4wDAYDVQQKEwVDaXNjbzAeFw0xODA3MjEyMDQzMjVa
> Fw00ODA3MTMyMDQzMjVaMIGIMS0wKwYDVQQDEyQxMmI0NGQ4Zi1kYWViLTRiMTYt
> YmFmYS1kYmE0MWRkZDFmNzMxCzAJBgNVBAYTAlVTMRMwEQYDVQQIEwpDYWxpZm9y
> bmlhMRAwDgYDVQQHEwdTYW5Kb3NlMRMwEQYDVQQLEwpBUElDRU0tU0ROMQ4wDAYD
> VQQKEwVDaXNjbzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALd/HK9i
> 6H45KJ42G3awGiehp5ZJWeUNqbmnkD0eiR6PNJIVNGBGMV3k+SW0kIfTjxIDkqSc
> K+K5tfEtTRc0R+WmgtdT+asDhPOmfmL73POqIXLr2RE98J0FLkcdhpRU2K1/tDu3
> 4+7WBA1OT/uIb9LVkVrk1pwolVKAWfgwBgCfmLmebtNfvexI3hvk2awcl4fmVb+d
> CMGVXHBw+yQW2AE3RH66VebS/E6bc3ifAvXHlWYWMt/tPRQRftH+0ZrgXt/dar5o
> 24Qq8taPwY1UIse41Nnnp1Q+FuoE9elLVmLidVaJ7qraDw5Yi6ZSQNGud813lU8s
> 6RtbNS48SSB3U0MCAwEAAaMdMBswDAYDVR0TBAUwAwEB/zALBgNVHQ8EBAMCAQYw
> DQYJKoZIhvcNAQELBQADggEBAHUddWTSeJR7QLxud9aaoIYzbL8lwJx7Ot49K5So
> RrymEKbrUSGtZaBp+hdam3l2ByDS44OmwDolelMsdIN61hV5o8DgpaJLwMhoY8gS
> HXyUTdpiLQD6RoxNWim6wGJGDTiMZFYaiaz5oFeazzOG7D8qu9agQ+/7Ky0RVkUK
> 6IoZZPSQ4BgBnsKg3OdBSPUWSIB7+rMz6ww5ELNOBIDo+yo0wCNgvmy19QgabM0/
> 5SJc2pMyidQy2yj3h1+DpMlZ6nWPqF+9IZlzoErKYABySH/Stho4l2kMXY5libCi
> AEqsyhYpxeO9S466uzcujQwJFARog2GGO0vJEDICqKm+lDI=
> -----END CERTIFICATE-----
>
nfvis#


nfvis#
nfvis# system certificate input filepath intdatastore:uploads/pnp_cert7.pem
Value for 'pem-data' (<string>):
[Multiline mode, exit with ctrl-D.]
> -----BEGIN CERTIFICATE-----
> MIICLDCCAdKgAwIBAgIBADAKBggqhkjOPQQDAjB9MQswCQYDVQQGEwJCRTEPMA0G
> A1UEChMGR251VExTMSUwIwYDVQQLExxHbnVUTFMgY2VydGlmaWNhdGUgYXV0aG9y
> aXR5MQ8wDQYDVQQIEwZMZXV2ZW4xJTAjBgNVBAMTHEdudVRMUyBjZXJ0aWZpY2F0
> ZSBhdXRob3JpdHkwHhcNMTEwNTIzMjAzODIxWhcNMTIxMjIyMDc0MTUxWjB9MQsw
> CQYDVQQGEwJCRTEPMA0GA1UEChMGR251VExTMSUwIwYDVQQLExxHbnVUTFMgY2Vy
> dGlmaWNhdGUgYXV0aG9yaXR5MQ8wDQYDVQQIEwZMZXV2ZW4xJTAjBgNVBAMTHEdu
> dVRMUyBjZXJ0aWZpY2F0ZSBhdXRob3JpdHkwWTATBgcqhkjOPQIBBggqhkjOPQMB
> BwNCAARS2I0jiuNn14Y2sSALCX3IybqiIJUvxUpj+oNfzngvj/Niyv2394BWnW4X
```

```
> uQ4RTEiywK87WRcWMGgJB5kX/t2no0MwQTAPBgNVHRMBAf8EBTADAQH/MA8GA1Ud
> DwEB/wQFAwMHBgAwHQYDVR0OBBYEFPC0gf6YEr+1KLlkQAPLzB9mTigDMAoGCCqG
> SM49BAMCA0gAMEUCIDGuwD1KPyG+hRf88MeyMQcqOFZD0TbVleF+UsAGQ4enAiEA
> l4wOuDwKQa+upc8GftXE2C//4mKANBC6It01gUaTIpo=
> -----END CERTIFICATE-----
>
nfvis# config
Entering configuration mode terminal
nfvis(config)# pnp automatic cco disable
nfvis(config)# pnp automatic cco-ipv6 disable
nfvis(config)# pnp automatic dns disable
nfvis(config)# pnp automatic dns-ipv6 disable
nfvis(config)# pnp automatic dhcp disable
nfvis(config)# pnp automatic dhcp-ipv6 disable
nfvis(config)# commit
Commit complete.
nfvis(config)# pnp static ip-address 10.0.0.7 port 443 transport https cafile
/data/intdatastore/uploads/pnp_cert7.pem
nfvis(config)# commit
Commit complete.
nfvis(config)# end
nfvis# exit
```

PnP Action

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

PnP Action API and Command

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

DPDK Support on NFVIS

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

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

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

DPDK support on NFVIS includes:

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

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

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

Restrictions

- You must enable DPDK using the **system settings dpdk enable** command before you commit any other configurations.
- Starting from Cisco NFVIS Release 4.12.1, you can employ DPDK with port-channels.
- DPDK is not supported on wan-br or wan2-br on any NFVIS platform.
- SR-IOV interfaces and DPDK support:

  To enable DPDK, every device driver must be supported by DPDK. NFVIS does not support SR-IOV interface upgrade to enable DPDK because SR-IOV device drivers are not supported by DPDK. If any SR-IOV network has been configured on an interface, that interface will not support DPDK. Also if an SR-IOV interface is attached to a bridge, the bridge does not support DPDK and if a bridge supports DPDK, no SR-IOV interface can be attached to it.

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

- VNF downtime:

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

System Requirements

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

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

Configuring DPDK Support on NFVIS

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

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

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

After enabling DPDK, physical NICs configured with SR-IOV will not be able to interact with DPDK bridges. To add a physical NIC to a DPDK bridge, all SR-IOV networks created on the interface should be removed first. NFVIS will not allow adding an SR-IOV configured interface to a DPDK bridge. For more information, see Dynamic SR-IOV, on page 58.

To enable DPDK support:

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

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

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

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

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

config terminal
no system settings dpdk
commit

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

Storage Access

Network File System Support

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.

Host System Operations

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

Power Cycle System

To power cycle NFVIS, use the following command:

```
nfvis# hostaction powercycle
```

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

Reboot System

To reboot NFVIS, use the following command:

```
nfvis# hostaction reboot
```

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

Shut down System

To shut down NFVIS, use the following command:

```
nfvis# hostaction shutdown
```

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

System file-list

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

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

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

System file-copy

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

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

The **system file-copy** command can also be used to copy a file from the given source path to the given destination path. The allowed directories for source path and destination path are:

- /data/intdatastore
- /mnt/extdatastore1
- /mnt/extdatastore2
- /data/mount

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

System file-delete

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

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

Secure Copy

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

The syntax for this command is:

scp *<source> <destination>*



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

SCP between two NFVIS devices is not supported.

Examples

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Change BIOS Password

It allows the user to change the BIOS password. A notification and syslog are sent regarding the password change.

To change the BIOS password:

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

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

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

Change CIMC Password

It allows the user to change the CIMC password. A notification and syslog are sent regarding the password change.

To change CIMC password:

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

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

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

Backup and Restore NFVIS and VM Configurations

Table 25: 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

Feature	NFVIS 4.1.1 and Earlier Releases	NFVIS 4.2.1 Release
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
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

	Backup configuration-only	Backup configuration-and-vms
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 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
```

APC UPS Support and Monitoring

Note This feature is supported only on ENCS 5400.

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

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

To enable APC UPS:

```
apcups enable
```

To disable APC UPS:

```
apcups disable
```

To check the battery status of an APC UPS:

```
apcups battery-status
```

Resetting to Factory Default

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

You can reset the host server to factory default with the following 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

Configure Banner, Message of the day and System Time

Configuring Your Banner and Message of the Day

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

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

To configure your banner and message:

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



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

Banner and Message APIs and Commands

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

Setting the System Time Manually or With NTP

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

To set the system time manually:

```
configure terminal
```

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



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

To set the system time using NTP IPv4:

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

To set the system time using NTP IPv6:

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

Verifying the System Time Configuration

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

```
nfvis# show system time

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

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

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

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

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

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

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

Configure DNS Name Servers

Table 26: Feature History

Feature Name	Release Information	Description
DNS Name Server Configuration Enhancement	NFVIS 4.4.1	You can now configure up to three name servers, which the DNS resolvers can use in the order you specify. New command introduced to configure DNS nameservers: **system settings name-server**.
Use **system settings name-server** command	NFVIS 4.10.1	Starting from Cisco NFVIS Release 4.10.1, the command **dns-server** is no longer supported. We recommend that you use **system settings name-server** command instead.

Restrictions

- In NFVIS 4.4 release, only **system settings name-server** command can be used for the configuration of DNS name servers at a given time.

This example shows how to configure name servers:

```
config terminal
system settings name-server 209.165.201.24 209.165.201.23 2001:420:30d:201:ffff:ffff:fff4:36
commit
```

This example shows how to unconfigure name servers:

```
config terminal
no system settings name-server
commit
```

This example shows how to update name servers:

```
config terminal
no system settings name-server
system settings name-server 209.165.201.23 2001:420:30d:201:ffff:ffff:fff4:33 209.165.201.27
commit
```

DNS name servers configured using the **system settings name-server** command is prepended to DNS name servers provided by a DHCP server automatically. To view the list of configured name servers, use the **show system settings-native dns** command.

To unconfigure name servers:

```
config terminal
no system settings name-server
commit
```

DNS Server APIs and Commands

DNS nameservers APIs	DNS nameserver commands
/api/config/system/settings/name-server	system settings name-server
/api/operational/system:system/settings-native/dns	show system settings-native dns

Configuring IP Host

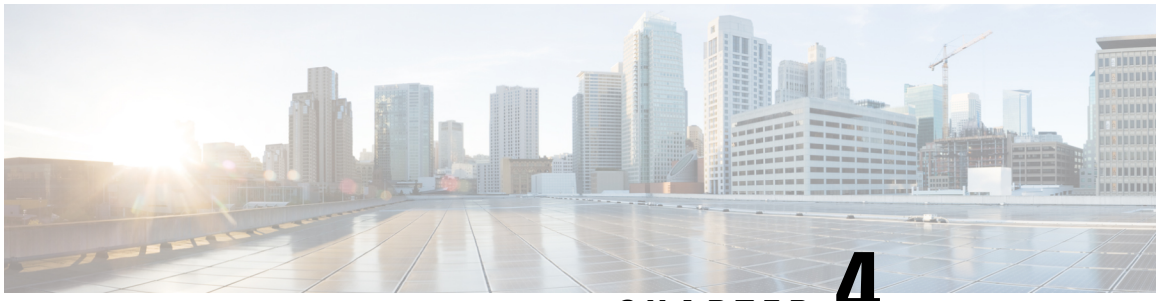
NFVIS IP host feature allows you to specify static mapping of host name and IP addresses.

To configure IP host mapping:

```
configure terminal
ip host test2.com 2.2.2.3 2.2.2.1
```

IP Host APIs and Commands

APIs	CLI Commands
/api/config/ip/host	ip host



CHAPTER 4

VM Life Cycle Management

Table 27: Feature History

Feature Name	Release Information	Feature Description
Virtual Machine CDROM Attachment and Detachment	Cisco NFVIS Release 4.18.1	The Day N CDROM Attachment and Detachment feature introduces new actions for virtual machines, enabling you to attach and detach ISO images. This functionality allows you to utilize an ISO image to troubleshoot, repair, or restore a VM's operating system.
Graceful VM Stop	Cisco NFVIS Release 4.18.1	The **Graceful Stop** feature provides an enhanced method for terminating virtual machine operations. Unlike forceful stop, this option sends a signal to the VM, allowing its services to shut down gracefully. The **Graceful stop** checkbox is enabled by default for VMs deployed as **OTHER** type when initiating a VM shutdown.

- Overview of VM Life Cycle Management, on page 90
- VM Image Packaging, on page 94
- Image Registration, on page 113
- VM Profiles or Flavors, on page 130
- Configure Internal Management Network, on page 130
- VM Deployment and Management, on page 133
- Access VNFs, on page 157
- Import and Export NFVIS VM, on page 159
- Secure Boot of VNFs, on page 162
- CDROM Attachment and Detachment, on page 163
- VM Graceful Stop, on page 164

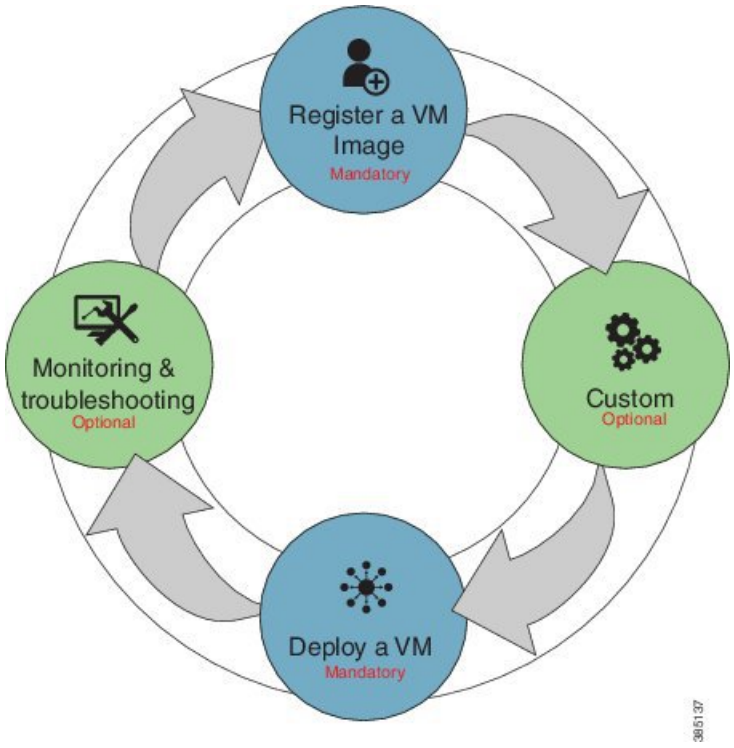
Overview of VM Life Cycle Management

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

Workflow of VM Life Cycle Management

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

Figure 1: VM Life Cycle Management



1. **Register a VM Image—**To 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*.

Support for NFVIS Container Lifecycle Management

Table 28: Feature History Table

Feature Name	Release Information	Description
Support for NFVIS Container Lifecycle Management	NFVIS 4.6	This features provides support for container lifecycle management. **Note** Cisco, may at its discretion, provide support (as defined by Cisco) to host container based applications for NFVIS 4.6.1. This capability is a beta feature in this release and should not be used in production because, Cisco may make changes in subsequent releases without providing any backward compatibility. The beta is provided AS-IS and without any warranty of any kind.

The NFVIS container lifecycle management follows the same process and data model as VM lifecycle management, with a few changes in the image registration and deployment process. For information about processes supported on container lifecycle mamagement, refer to:

- NFVIS Container Image Registration
- NFVIS Container Deployment

Uploading VM Images to an NFVIS Server

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

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

To copy an image using the USB device:

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



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

Verifying the Image Copied from the USB Drive

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

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

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

2 escmanager-tagged-log.log-20170314.gz /data/intdatastore/logs/2017-03/14/10-00 167K Other
 2017-01-18 05:58:26
3 confd_audit.log-20170317.gz /data/intdatastore/logs/2017-03/17/09-30 4.6K Other 2017-03-17
 21:29:59
4 esc_postinit.log-20170317.gz /data/intdatastore/logs/2017-03/17/05-00 605K Other 2017-03-17
 16:40:19
5 error.log-20170317.gz /data/intdatastore/logs/2017-03/17/05-00 1.3K Other 2017-03-17
16:40:15
6 ovs-ctl.log-20170317.gz /data/intdatastore/logs/2017-03/17/12-00 20 Other 2017-03-16
00:00:01 4:01
!
!
!
62 ovs-ctl.log-20170323.gz /data/intdatastore/logs/2017-03/23/12-00 20 Other 2017-03-22
00:00:01
63 CentOS-7-x86_64-Everything-1511.ova /data/intdatastore/uploads 1.1G VM 2017-03-15 19:20:03
 Package
64 TinyLinux.tar.gz /data/intdatastore/uploads 17M VM 2017-03-15 18:25:00 Package
65 Cisco-KVM-vWAAS-1300-6.3.0-b98.tar.gz /data/intdatastore/uploads 979M VM 2017-03-15
19:19:11 Package
```

```
66 ubuntu_14.04.3-server-amd64-disk1.tar /data/intdatastore/uploads 527M VM 2017-03-15
19:20:17.gz Package
67 asav961.tar.gz /data/intdatastore/uploads 164M VM 2017-03-15 18:24:57 Package
68 isrv-universalk9.16.03.01.tar.gz /data/intdatastore/uploads 1.3G VM 2017-03-15 19:19:53
```

Related APIs and Commands

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

Performing Resource Verification

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

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

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

CPU Over-Subscription

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

Configuring Management IP Subnet

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

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

To create a new subnet:

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

VM Image Packaging

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

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

VM Image Packaging Utility

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

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

Contents

The VM image packaging utility contains the following:

- nfvpt.py—It is a python based packaging tool that bundles the VM raw disk image/s along with VM specific properties.

- image_properties_template.xml—This is the template file for the VM image properties file, and has the parameters with default values. If the user provides new values to these parameters while creating the VM package, the default values get replaced with the user-defined values.
- 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 29: 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.

Parameter	Mandatory/Optional	Description
package_file_name	Mandatory	File name for the target VNF package. The default is the root disk image name with extension *.tar.gz*.
disk_img_names	Mandatory	List of root disk images to be bundled. Only the qcow2 images are supported.
img_name	Mandatory	Name of the VNF image.
vnf_type	Mandatory	VNF type Supported types are: ROUTER, FIREWALL, vWAAS, vWLC, and OTHER.
vnf_version	Mandatory	VNF version
monitored	Mandatory	VM health monitoring for those VMs that can be bootstrapped Options are: true/false Monitoring timeout period for a monitored VM is 600 seconds by default
optimize	Mandatory	Optimized VM Options are: true/false
virtual_interface_model	Optional	Default is none.
thick_disk_provisioning	Optional	Default is false.
eager_zero	Optional	Default is false.
bootstrap_cloud_init_bus_type	Optional	Default is IDE.
bootstrap_cloud_init_drive_type	Optional	Mounts the day0 configuration file as disk Default is CD-ROM.
bootstrap	Optional	Bootstrap files for VNF. Two parameters are required in the format of dst:src; dst filename including path has to match exactly to what the VM expects; up to 20 bootstrap files are accepted. For example: --bootstrap ovf-env.xml for ISRv and --bootstrap day0-config for ASAv.
min_vcpu	Optional	Minimum number of vCPUs supported by the VM. The default is 1.

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

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

NFVIS Specific Enhancements

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

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

```
 --pack_dir <DIR> PACK

                       package all files in directory

Resources:

  --vnic_names VNIC_NAMES

                       list of vnic number to name mapping in format

                       number:name example --vnic_names

                       1:GigabitEthernet2,2:GigabitEthernet4
```

Usage

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

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

VM Packaging Utility Usage Examples

Given below are the contents of the file *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 30: 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 31: 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 32: VM Image Properties File Details

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

Disk-1 bus type	Additional disk1 image disk bus	<disk_1_file_disk_bus>	virtio, scsi, and ide	Optional
Disk-2 bus type	Disk2 image disk bus	<disk_2_file_disk_bus>	virtio, scsi, and ide	Optional
Disk-10 bus type	Disk10 image disk bus	<disk_10_file_disk_bus>	virtio, scsi, and ide	Optional
Root Disk Image format	Root image disk format	<root_image_disk_format>	qcow2 and raw	Mandatory
Disk-1 Image format	Additional disk 1 image format	<disk_1_image_format>	qcow2 and raw	Optional
Disk-2 Image format	Disk 2 image format	<disk_2_image_format>	qcow2 and raw	Optional
Disk-10 Image format	Disk 10 image format	<disk_10_image_format>	qcow2 and raw	Optional
Serial Console	Serial console supported	<console_type_serial>	true, false	Optional
Minimum vCPU	Minimum vCPUs required for a VM operation	<vcpu_min>		Mandatory
Maximum vCPU	Maximum vCPUs supported by a VM	<vcpu_max>		Mandatory
Minimum memory	Minimum memory in MB required for VM operation	<memory_mb_min>		Mandatory
Maximum memory	Maximum memory in MB supported by a VM	<memory_mb_max>		Mandatory
Minimum root disk size	Minimum disk size in GB required for VM operation	<root_disk_gb_min>		Optional
Maximum root disk size	Maximum disk size in GB supported by a VM	<root_disk_gb_max>		Optional
Maximum vNICs	Maximum number of vNICs supported by a VM	<vnic_max>		Mandatory

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

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

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

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



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

Example: Package.mf

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

Example: Image Properties

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

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

Example: Bootstrap Configuration File

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

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

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

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

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

Image Properties Template File

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

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

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

Image Registration

Table 33: Feature History

Feature Name	Release Information	Description
VM Image Registration using FTP and SCP servers	Cisco NFVIS Release 4.12.1	In addition to the already existing support to register a VM image using a http or https servers, you can register the VM images using FTP and SCP servers.

To register a VM image, you must first copy or download the relevant VM image to the NFVIS server, or host the image on a http or https or FTP or SCP servers. Once you have downloaded the file, you can register the image using the registration API. The registration API allows you to specify the file path to the location (on the http/https/ftp/scp server) where the tar.gz file is hosted. Registering the image is a one-time activity. Once an image is registered on the http or https server, and is in active state, you can perform multiple VM deployments using the registered image. All VM images are available in VM packaging and VM package content. For more information see, VM Image Packaging Utility, on page 94

Here are some of the example configurations to register a VM image using the supported servers:

Remote Server Configuration Example

Here is a sample configuration to configure a remote server:

```
<remote_servers>
    <remote_server>
        <name>myserver</name>
        <base_url>ENTER_IP_ADDRESS</base_url>
        <username>ENTER_USERNAME</username>
        <password>ENTER_PASSWORD_HERE</password>
    </remote_server>
</remote_servers>
```

Register Images Using the SCP Server

Here is a sample configuration to register images using the SCP server:

```
<images>
    <image>
        <name>C8Kv_IMAGE</name>
       <src>scp://myserver/nfvis/c8kv.tar.gz>  <============ Specify the remote server
 name created in device
        <remove_src_on_completion>
true</remove_src_on_completion>
    </image>
</images>
```

Register Images Using the FTP Server

Here is a sample configuration to register images using the FTP server:

```
<images>
    <image>
        <name>C8Kv_IMAGE</name>
        <src>ftp://myserver/nfvis/c8kv.tar.gz</src>   <============ Specify the remote
server name created in device
        <remove_src_on_completion>
true</remove_src_on_completion>
    </image>
</images>
```

Register Images Using the HTTP/HTTPS Server

Here is a sample configuration to register images using the HTTP/HTTPS server:

```
<images>
    <image>
        <name>C8Kv_IMAGE</name>
        <src>http://myserver/nfvis/c8kv.tar.gz</src>    <============ Specify the remote
server name created in device
        <remove_src_on_completion>
true</remove_src_on_completion>
    </image>
</images>
```

Register VM Packages Using REST API

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

Post Image Registration

```
curl -v -u -k admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml
-X POST https://209.165.201.1 /api/config/vm_lifecycle/images -d
'<image><name>WinServer2012R2.iso</name><src>file:///data/intdatastore/uploads/WinServer2012R2.iso/WinServer2012R2.iso</src></image>'

HTTP/1.1 201 Created
```

Get Image Status

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X
GET https://209.165.201.1/api/operational/vm_lifecycle/opdata/images/image/isrv-03.16.02?deep

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

Get Registered Image Status

```
Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X
```

```
GET https://209.165.201.1/api/config/vm_lifecycle/images?deep

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

Delete Registered Image

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X
DELETE https://209.165.201.1/api/config/vm_lifecycle/images/image/isrv-3.16.0.1a

HTTP/1.1 204 No Content
```

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

Register VM Image with Multiple Root Disks

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

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

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

Note Image profiles apply to the first root disk only. The size of the remaining disks remain the same when deployed through different profiles with an exception to vWAAS VNF.

Register a VM Image through a Root Disk

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

Default Properties for Root Disk Registration

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

Property Name	Property Tag	Default Value
Version	<version>	NA
VNF Type	<vnf_typ>	OTHER
VCPU Min	<vcpu_min>	1
VCPU Max	<vcpu_max>	64
Memory Min (MB)	<memory_mb_min>	256
Memory Max (MB)	<memory_mb_max>	1048576
Root Disk Min Size (GB)	<root_disk_gb_min>	1
Root Disk Max Size (GB)	<root_disk_gb_max>	10240
VNIC Max	<vnic_max>	8
Bootup Time	<bootup_time>	-1
Interface Hot Add	<interface_hot_add>	true
Interface Hot Delete	<interface_hot_delete>	false

Register a Remote VM Image

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

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

Example: POST Remote Image Registration from Webserver over HTTPS

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

HTTP/1.1 201 Created
```

Example: POST Remote Image Registration from Webserver over HTTPS

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

```
 <name>ASAV</name>
 <src>https://172.20.117.124/nfvis/asav982.tar.gz</src>
 <certificate_validation>true</certificate_validation>
 <certificate_file>/data/intdatastore/uploads/pub_key.cert</certificate_file>
</image>'

HTTP/1.1 201 Created
```

Specify Storage Location for a VM Image

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

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

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



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

Example: VM Image Storage Placement

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

HTTP/1.1 201 Created
```

Update VM Image

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

- interface_hot_add
- interface_hot_delete

Starting from NFVIS 4.1 release, interface hot add and delete are now supported on DPDK networks.

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

Example: Delete Value and Add New Value

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

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

HTTP/1.1 204 No Content
```

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

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

HTTP/1.1 201 Created
```

Example: Get All Properties and Values

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

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

Image Properties

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

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

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

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

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

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
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

Property Name	Description	Property Tag	Possible Values	Mandatory/ 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

Property Name	Description	Property Tag	Possible Values	Mandatory/ 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

Property Name	Description	Property Tag	Possible Values	Mandatory/ 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
Disable Spoof Check	Used to disable spoof check for Privledged VMs	<disable_spoof_check>	true, false	Optional
Virtual interface model		<virtual_interface_model>		Optional

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
Interface Hot Add	If true, an active VNF's virtual interface can be added/updated without shutting down the VNF.	<interface_hot_add>	true, false	Optional, default true
Interface Hot Delete	If true, an active VNF's virtual interface can be deleted without shutting down the VNF.	<interface_hot_delete>	true, false	Optional, default false
Thick disk provisioning	During deployment, VM will be a fully allocated root disk with size specified by flavor.	<thick_disk_provisioning>	true, false	Optional, default false
Eager Zero	Used in conjunction with Thick disk provisioning. During deployment, root disk is zeroed out to improve I/O operations	<eager_zero>	true, false	Optional, only valid if Thick disk provisioning is enabled. Default false
Profile for VM deployment	A profile defines the resources required for VM deployment. This profile is referenced during VM deployment.	<profile>		Optional
Name	Profile name	<name>	Any	Mandatory
Description	Description of the profile	<description>	Any	Mandatory

Property Name	Description	Property Tag	Possible Values	Mandatory/ Optional
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

Example: Contents of an image_properties.xml File

```
<?xml version="1.0" encoding="UTF-8"?>
<image_properties>
    <vnf_type>ROUTER</vnf_type>
    <name>ISRV</name>
    <version>16.06.05</version>
    <bootup_time>600</bootup_time>
    <root_file_disk_bus>virtio</root_file_disk_bus>
    <root_image_disk_format>qcow2</root_image_disk_format>
    <vcpu_min>1</vcpu_min>
    <vcpu_max>8</vcpu_max>
    <memory_mb_min>4096</memory_mb_min>
    <memory_mb_max>8192</memory_mb_max>
    <vnic_max>8</vnic_max>
    <vnic_names>vnics:1:GigabitEthernet2</vnic_names>
    <vnic_names>vnics:2:GigabitEthernet3</vnic_names>
    <vnic_names>vnics:3:GigabitEthernet4</vnic_names>
    <vnic_names>vnics:4:GigabitEthernet5</vnic_names>
    <vnic_names>vnics:5:GigabitEthernet6</vnic_names>
```

```
    <vnic_names>vnics:6:GigabitEthernet7</vnic_names>
    <vnic_names>vnics:7:GigabitEthernet8</vnic_names>
    <root_disk_gb_min>8</root_disk_gb_min>
    <root_disk_gb_max>8</root_disk_gb_max>
    <console_type_serial>true</console_type_serial>
    <sriov_supported>true</sriov_supported>
    <sriov_driver_list>igb</sriov_driver_list>
    <sriov_driver_list>igbvf</sriov_driver_list>
    <sriov_driver_list>i40evf</sriov_driver_list>
    <pcie_supported>true</pcie_supported>
    <pcie_driver_list>igb</pcie_driver_list>
    <pcie_driver_list>igbvf</pcie_driver_list>
    <pcie_driver_list>i40evf</pcie_driver_list>
    <monitoring_supported>true</monitoring_supported>
    <monitoring_methods>ICMPPing</monitoring_methods>
    <low_latency>true</low_latency>
    <privileged_vm>true</privileged_vm>
    <cdrom>true</cdrom>
    <bootstrap_file_1>ovf-env.xml</bootstrap_file_1>
    <bootstrap_file_2>iosxe_config.txt</bootstrap_file_2>
    <custom_property>
        <tech_package>ax</tech_package>
        <tech_package>security</tech_package>
        <tech_package>ipbase</tech_package>
        <tech_package>appx</tech_package>
    </custom_property>
    <custom_property>
        <ngio>enable</ngio>
    </custom_property>
    <custom_property>
        <SSH_USERNAME> </SSH_USERNAME>
    </custom_property>
    <custom_property>
        <SSH_PASSWORD> </SSH_PASSWORD>
    </custom_property>
    <profiles>
     <profile>
     <name>ISRv-mini</name>
            <description>ISRv-mini</description>
     <vcpus>1</vcpus>
            <memory_mb>4096</memory_mb>
            <root_disk_mb>8192</root_disk_mb>
 </profile>
     <profile>
     <name>ISRv-small</name>
            <description>ISRv-small</description>
     <vcpus>2</vcpus>
            <memory_mb>4096</memory_mb>
            <root_disk_mb>8192</root_disk_mb>
 </profile>
     <profile>
     <name>ISRv-medium</name>
            <description>ISRv-medium</description>
     <vcpus>4</vcpus>
            <memory_mb>4096</memory_mb>
            <root_disk_mb>8192</root_disk_mb>
 </profile>
    </profiles>
    <default_profile>ISRv-small</default_profile>
</image_properties>
```

NFVIS Container Image Registration

The NFVIS container image registration follows the same workflow as VM image registration. However, there is a difference in the container image registration payload, where a **<locator>** field is included. This field must be populated with the word "container", and it signifies that it is a container image and not a VM image. Only files with the extension .docker are supported and can be uploaded for container image registration.

Here's a sample payload of container image registration:

```
<vm_lifecycle xmlns:ns2="urn:ietf:params:xml:ns:netconf:notification:1.0"
xmlns:ns1="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:ns3="http://www.cisco.com/esc/esc_notifications"
xmlns:ns0="http://www.cisco.com/nfvis/vm_lifecycle"
xmlns="http://www.cisco.com/nfvis/vm_lifecycle">
<images>
    <image>
        <name>ubuntu-1</name>
        <src>file:///data/intdatastore/uploads/ubuntu.docker</src>
        <remove_src_on_completion>true</remove_src_on_completion>
        <locator>
            <vim_id>container</vim_id>
           </locator>
    </image>
</images>
</vm_lifecycle>
```

Note The container image does not include any profiles. Before deploying a container, you have to either create a customized profile or use a profile from another image that is suitable.

Image Registration for CUCM Applications

Collaboration applications such as Cisco Unified Communications Manager (CUCM), Cisco Unity Connection, and Cisco IM and Presence Service (IM&P) require registering, both together, the ISO image (installation media), and the OVA file (metadata) with NFVIS.

Local Image Registration

Use this method when the ISO and OVA files are available on your local computer.

1. Navigate to the Image Repository.

   Access the image repository via **Configuration** > **Virtual Machine** > **Images** > **Image Repository**

2. Upload the ISO image.
   a. Click **Upload**.
   b. Select the ISO file from your local file system.

      Example: Bootable_UCSInstall_UCOS_15.0.1.iso

3. Configure image properties.
   a. Update the pre-populated image name to a descriptive name.

Example: cucm-15.0.1

b. Select the appropriate **VM Type** from the drop-down list. If your application type is not listed, select **Other**.

4. Upload the OVA metadata.

a. Select the **Metadata** check box.

b. Click **Upload OVA** and select the OVA file.

Example: cucm_15.0_vmv17_v1.2.sha512.ova

5. Configure additional options.

Select the **Dedicated Cores** check box to allocate dedicated CPU cores for better performance.

6. Submit the image registration request.

Click **Upload File** to submit the request.

NFVIS uploads the ISO, uploads and parse the OVA, register the image, and automatically create profiles and flavors from the OVA metadata.

7. Verify successful registration.

a. Wait for the registration to complete.

Confirm the image appears in the Image Repository with an **Active** status.

b. Verify that flavors are populated from the OVA file.

Remote Image Registration

Use this method when files are hosted on a remote server (HTTP, HTTPS, FTP, or SCP).

Table 34: Protocols

Protocol	Authentication	Notes
HTTP	Not supported	No Username and password
HTTPS	Not supported	No Username and password
FTP	Required	Username and password
SCP	Required	Username and password

1. Navigate to the Image Repository.

Access the image repository via **Configuration** > **Virtual Machine** > **Images** > **Image Repository**

2. Select the remote registration option.

a. Click **Register Image**.

b. Select **Remote Image Registration**.

Example: cucm-15.0.1

3. Configure image properties.
   a. Enter a descriptive name in the **Image name** field.
4. Configure the remote server connection.
   a. Select the **Protocol** from the drop-down list (HTTP, HTTPS, FTP, or SCP).
   b. Enter the server hostname or IP address in the **IP Address** field.
   c. Enter the directory path to the ISO file in the **Image File Path** field.

Table 35: Remote Connection Examples

Protocol	IP address	Image File Path
HTTP	192.168.1.100	/images/Bootable_UCSInstall_UCOS_15.0.1.iso
FTP	192.168.1.100	/pub/cisco/Bootable_UCSInstall_UCOS_15.0.1.iso
SCP	192.168.1.100	/home/user/images/Bootable_UCSInstall_UCOS_15.0.1.is

5. Configure authentication for FTP or SCP protocols.
   a. If you selected FTP or SCP as the protocol, enter the Username for the remote server.
   b. Enter the Password.

      Authentication is not supported for HTTP or HTTPS protocols.
6. Upload the OVA metadata.
   a. Select the **Metadata** check box.
   b. Enter the path to the OVA file in the **OVA File Path** field.

      Example: /home/user/images/cucm_15.0_vmv17_v1.2.sha512.ova

      The OVA file must reside on the same remote server as the ISO file.
7. Configure additional options.

   Select the **Dedicated Cores** check box if dedicated CPU cores are required for the virtual machine.
8. Click **Submit** to submit the registration request.

   NFVIS downloads the ISO image and OVA file from the remote server, parses the OVA file, registers the image, and automatically creates profiles and flavors from the OVA metadata.
9. Monitor the registration progress.

   Track the registration progress in the Image Repository. Downloading large files can take several minutes.

VM Profiles or Flavors

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

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



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

Example: Create VM Profile Using Rest API

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

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X POST
https://209.165.201.1/api/config/vm_lifecycle/flavors -d '<flavor>
    <name>windows</name>
    <ephemeral_disk_mb>0</ephemeral_disk_mb>
    <memory_mb>4096</memory_mb>
    <root_disk_mb>12288</root_disk_mb>
    <swap_disk_mb>0</swap_disk_mb>
    <vcpus>2</vcpus>
</flavor>'
```

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

Configure Internal Management Network

Table 36: Feature History Table

Feature Name	Release Information	Description
DHCP Support for Internal Management Network.	Cisco NFVIS 4.12.1	Cisco NFVIS uses the IP address assigned by a DHCP server for monitoring the VNF deployment.

The internal management network is a separate network within the Cisco NFVIS infrastructure that is used for VM monitoring and recovery. The internal management network is used to handle communication between the Cisco NFVIS and other management-related functions like port forwarding. Cisco NFVIS creates an internal management network named **int-mgmt-net** by default.

Here's an example of the default int-mgmt-net configuration:

```
vm_lifecycle networks network int-mgmt-net
 subnet int-mgmt-net-subnet
  address 10.20.0.0
```

```
  netmask 255.255.255.0
  gateway 10.20.0.1
 !
!
```

You can update the default configuration before deploying VMs/VNFs. When the default config is deleted, Cisco NFVIS recreates the internal management network upon rebooting the host.

Deploy VNF Using Internal Management Network

Cisco NFVIS lets you configure the VNF deployment to be monitored. To monitor the VM deployment, an interface is attached to the internal management network (int-mgmt-net). The interface attached contains the NIC ID **0** by default.

Examples of Attaching an Interface to an Internal Management Network

Internal Management Network in the VM Deployment Payload	IP address token/variable in Day-0 config	Behavior

Example configuration using IP address:
`<interface>`
`<nicid>0</nicid>`
`<network>int-mgmt-net</network>`
`<ip_address>10.20.0.21</ip_address>`
`</interface>`
	Token ${NICID_0_IP_ADDRESS}" is either present or absent.	If the IP payload is available, the payload is assigned to the interface.
		If there is a token/variable, the IP address is replaced with the token.
		When the IP address in the IP payload is unavailable or incorrect, the deployment fails.

Example configuration when no IP address is configured:
`<interface>`
`<nicid>0</nicid>`
`<network>int-mgmt-net</network>`
`</interface>`
	${NICID_0_IP_ADDRESS} token is present in the day-0 config.	Cisco NFVIS auto assigns the IP address and the token/variable is replaced in the day-0 config with the same IP address.

Example configuration when no IP address is configured with the NIC ID being absent:
`<interface>`
`<nicid>0</nicid>`
`<network>int-mgmt-net</network>`
`</interface>`
	Not present
	By default, the interface is configured as DHCP. The token is not replaced.
		Note
		Starting from Cisco NFVIS Release 4.12.1, configuring an interface as a DHCP server is supported.
		The IP address is auto allocated using the internal DHCP server. Cisco NFVIS utilizes the IP address for monitoring the deployment.

Here is a sample day-0 config with the IP address and the token/variable:

```
<?xml version="1.0" encoding="UTF-8"?>
<Environment
xmlns:oe="http://schemas.dmtf.org/ovf/environment/1">
  <PropertySection>
    <Property oe:key="com.cisco.c8000v.config-version.1" oe:value="1.0"/>
    <Property oe:key="com.cisco.c8000v.enable-ssh-server.1" oe:value="True"/>
```

```
    <Property oe:key="com.cisco.c8000v.login-username.1" oe:value="${SSH_USERNAME}"/>
    <Property oe:key="com.cisco.c8000v.login-password.1" oe:value="${SSH_PASSWORD}"/>
    <Property oe:key="com.cisco.c8000v.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.c8000v.mgmt-ipv4-addr.1"
oe:value="${NICID_0_IP_ADDRESS}/${NICID_0_CIDR_PREFIX}"/>         ===> IP address set via
token
    <Property oe:key="com.cisco.c8000v.mgmt-ipv4-network.1" oe:value=""/>
    <Property oe:key="com.cisco.c8000v.license.1" oe:value="${TECH_PACKAGE}"/>
   <Property oe:key="com.cisco.c8000v.ios-config-0001" oe:value="vrf definition Mgmt-intf"/>

    <Property oe:key="com.cisco.c8000v.ios-config-0002" oe:value="address-family ipv4"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0003" oe:value="exit-address-family"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0004" oe:value="address-family ipv6"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0005" oe:value="exit-address-family"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0006" oe:value="exit"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0007" oe:value="interface
GigabitEthernet1"/>
   <Property oe:key="com.cisco.c8000v.ios-config-0008" oe:value="vrf forwarding Mgmt-intf"/>

    <Property oe:key="com.cisco.c8000v.ios-config-0009" oe:value="ip address
${NICID_0_IP_ADDRESS} ${NICID_0_NETMASK}"/>   ===> IP address set via token
    <Property oe:key="com.cisco.c8000v.ios-config-0010" oe:value="no shut"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0011" oe:value="exit"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0012" oe:value="ip route vrf Mgmt-intf
0.0.0.0 0.0.0.0 ${NICID_0_GATEWAY}"/>
  </PropertySection>
</Environment>
```

Here is a sample day-0 config using DHCP server:

```
<?xml version="1.0" encoding="UTF-8"?>
<Environment
xmlns:oe="http://schemas.dmtf.org/ovf/environment/1">
  <PropertySection>
    <Property oe:key="com.cisco.c8000v.config-version.1" oe:value="1.0"/>
    <Property oe:key="com.cisco.c8000v.enable-ssh-server.1" oe:value="True"/>
    <Property oe:key="com.cisco.c8000v.login-username.1" oe:value="${SSH_USERNAME}"/>
    <Property oe:key="com.cisco.c8000v.login-password.1" oe:value="${SSH_PASSWORD}"/>
    <Property oe:key="com.cisco.c8000v.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.c8000v.mgmt-ipv4-addr.1" oe:value=""/>
                ===> IP address left blank here
    <Property oe:key="com.cisco.c8000v.mgmt-ipv4-network.1" oe:value=""/>
    <Property oe:key="com.cisco.c8000v.license.1" oe:value="${TECH_PACKAGE}"/>
   <Property oe:key="com.cisco.c8000v.ios-config-0001" oe:value="vrf definition Mgmt-intf"/>

    <Property oe:key="com.cisco.c8000v.ios-config-0002" oe:value="address-family ipv4"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0003" oe:value="exit-address-family"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0004" oe:value="address-family ipv6"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0005" oe:value="exit-address-family"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0006" oe:value="exit"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0007" oe:value="interface
GigabitEthernet1"/>
   <Property oe:key="com.cisco.c8000v.ios-config-0008" oe:value="vrf forwarding Mgmt-intf"/>

   <Property oe:key="com.cisco.c8000v.ios-config-0009" oe:value="ip address dhcp setroute"/>
              ===> DHCP specified vs IP address
    <Property oe:key="com.cisco.c8000v.ios-config-0010" oe:value="no shut"/>
    <Property oe:key="com.cisco.c8000v.ios-config-0011" oe:value="exit"/>
  </PropertySection>
</Environment>
```

VM Deployment and Management

VM Deployment

A VM can be deployed through API or CLI. Ensure that you have registered a VM image before attempting to deploy it. For more details, see Image Registration, on page 113.

Note The VM name must meet the following requirements:

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

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

Example: Deploy VMs Using REST API

Use the following API to deploy a VM.

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

Here's a sample payload of deploying a VM.

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X POST
https://209.165.201.1/api/config/vm_lifecycle/tenants /tenant/admin/deployments --data '
<deployment>
  <name>ISRdep</name>
  <vm_group>
    <name>ISRvmgrp</name>
    <image>ISR_IMAGE</image>
    <bootup_time>500</bootup_time>
    <recovery_wait_time>0</recovery_wait_time>
    <interfaces>
      <interface>
        <nicid>0</nicid>
        <network>int-mgmt-net</network>
        <ip_address>10.20.0.21</ip_address>
        <port_forwarding>
          <port>
            <type>ssh</type>
            <protocol>tcp</protocol>
            <vnf_port>22</vnf_port>
            <external_port_range>
              <start>20022</start>
              <end>20022</end>
            </external_port_range>
```

```
          </port>
        </port_forwarding>
      </interface>
    </interfaces>
    <kpi_data>
      <kpi>
        <event_name>VM_ALIVE</event_name>
        <metric_value>1</metric_value>
        <metric_cond>GT</metric_cond>
        <metric_type>UINT32</metric_type>
        <metric_collector>
          <type>ICMPPing</type>
          <nicid>0</nicid>
          <poll_frequency>3</poll_frequency>
          <polling_unit>seconds</polling_unit>
          <continuous_alarm>false</continuous_alarm>
        </metric_collector>
      </kpi>
    </kpi_data>
    <rules>
      <admin_rules>
        <rule>
          <event_name>VM_ALIVE</event_name>
          <action>ALWAYS log</action>
          <action>TRUE servicebooted.sh</action>
          <action>FALSE recover autohealing</action>
        </rule>
      </admin_rules>
    </rules>
    <config_data>
      <configuration>
        <dst>bootstrap_config</dst>
        <variable>
          <name>TECH_PACKAGE</name>
          <val>ax</val>
        </variable>
      </configuration>
    </config_data>
  </vm_group>
</deployment>'
```

Verify VM Deployment

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

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

```
nfvis# show vm_lifecycle opdata tenants tenant admin deployments ROUTER
deployments ROUTER
 deployment_id SystemAdminTenantIdROUTER
 vm_group ROUTER
  bootup_time 600
  vm_instance d1c462e9-2706-4868-befd-d8f7806b9444
   name     ROUTER
   host_id  NFVIS
   hostname nfvis
   interfaces interface 0
    model       virtio
    type        virtual
    port_id     vnic0
    network     int-mgmt-net
    subnet      N/A
```

```
    ip_address  10.20.0.2
    mac_address 52:54:00:fe:34:53
    netmask     255.255.255.0
    gateway     10.20.0.1
   interfaces interface 1
    type        virtual
    port_id     vnic1
    network     GE0-1-SRIOV-1
    subnet      N/A
    mac_address 52:54:00:e4:13:67
 state_machine state SERVICE_ACTIVE_STATE
VM
NAME    STATE
-----------------------
ROUTER  VM_ALIVE_STATE
```



Note If the deployment was accepted first but failed later, NFVIS sends a notification with error status and message. Also on VM deployment failure, the operational data may or may not show the complete data about the failed VM.

VM Deployment Parameters

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

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

`bootup_time`	This parameter is no longer mandatory from 3.12 and later, provided that it is specified in image properties. **Accepted Values:** • For unmonitored VMs: -1 • For monitored VMs: Number of seconds	`<bootup_time>500</bootup_time>`
`vim_vm_name`	Optional. If a custom VM name is provided at the time of deployment, it may be used for all commands that accept VM name as an input.	
`kpi_data`	Mandatory for monitored VMs.	
`rules`	Mandatory for monitored VMs.	

`config_data`	Mandatory if the day-0 configuration has variables that have tokens assigned to them.	

`Encrypted config variable`	Optional.
	Only one value for a variable is allowed to be encrypted.
		`<variable>`
		`<name>TEST_VARIABLE</name>`
		`<encrypted_val>test_value</encrypted_val>`
		`</variable>`

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

volumes	Optional. Up to 2 volumes could be added to a deployment. Location of the volumes can be local or NFS (needs NFS mount name to be specified in case of NFS)	
port_forwarding	Optional. If port forwarding is included, all elements under it are mandatory.	<port_forwarding> <port> <type>ssh</type> <protocol>tcp</protocol> <vnf_port>22</vnf_port> <external_port_range> <start>20022</start> <end>20022</end> </external_port_range> </port> </port_forwarding>
Ngio interface	Optional. Used in config_data. To enable NIM support on a Cisco ISRv running on Cisco ENCS, you must use the variables in the ISRv deployment payload.	<variable> <name>ngio</name> <val>enable</val> </variable>

`interface model`	Optional. If the model is not specified for an interface, the default model is used. For Windows, the default model is rtl8139.	`<interface><nicid>3</nicid><network>wan-net</network><model>virtio</mode`
`vnc`	Optional. If the VNC password is not specified, there is no default password.	`<vnc> <password>vnc_password</password></vnc>`

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.

AutoVF Support

Table 37: Feature History

Feature Name	Release Information	Description
AutoVF Support	Cisco NFVIS Release 4.12.1	You no longer have to specify the Virtual Function (VF) ID of the SR-IOV interface during deployment or specify the VLAN information in the network namespace. Instead, you provide the PNIC name of the SR-IOV enabled PNIC and Cisco NFVIS auto-allocates an available VF on the SR-IOV PNIC.

To attach an SR-IOV or host interface to the VM interface, specify the PNIC name of the SR-IOV or host interface in the configuration. The below configuration is an example of how to specify a PNIC name:

```
<vm_lifecycle>
...
<interface>
    <nicid>1</nicid>
    <model>virtio</model>
    <pnic>GE0-0</pnic>
    <vlan>25</vlan>
    <trunk>false</trunk>
</interface>
...
</vm_lifecycle>
```

Note

- Use **vlan** to specify a VLAN tag for an interface. You can use **vlan** only if **trunk** is set to **false** or else the PNIC allocation fails
- Set **trunk** to **true** only to enable the interface with the trunk mode. The trunk mode allows packets of any VLAN tag to pass through the interface.

For more information see, vm_lifecycle tenants tenant admin deployments section of the VM Lifecycle Management Commands.

VM Monitoring

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

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

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

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

VNF Deployment Placement

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



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

Example: VM Deployment Using Placement

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X POST
https://209.165.201.1/api/config/vm_lifecycle/tenants
/tenant/admin/deployments --data
'<deployment>
<name>WINIsodep</name>
<vm_group>
  <name>WINIsovmgrp</name>
  <image>WinServer2012R2.iso</image>
  <flavor>windows</flavor>
  <bootup_time>-1</bootup_time>
  <recovery_wait_time>0</recovery_wait_time>
  <kpi_data>
   <enabled>true</enabled>
  </kpi_data>
   <scaling>
   <min_active>1</min_active>
   <max_active>1</max_active>
   <elastic>true</elastic>
   </scaling>
   <placement>
   <type>zone_host</type>
   <enforcement>strict</enforcement>
   <host>datastore1</host>
   </placement>
   <recovery_policy>
   <recovery_type>AUTO</recovery_type>
   <action_on_recovery>REBOOT_ONLY</action_on_recovery>
   </recovery_policy>
</vm_group>
</deployment>'
```

VM Export with Selective Disk

Table 38: Feature History

Feature Name	Release Information	Description
VM Export with Selective Disk	NFVIS 4.4.1	You can exclude certain disks or volumes from a VM export.

Starting from NFVIS 4.4, you can exclude certain disks or volumes from a VM export.

```
nfvis# show running-config vm_lifecycle tenants tenant admin deployments
vm_lifecycle tenants tenant admin
 deployments deployment linuxdep
  vm_group gp_01
   image              Linux_IMAGE
   flavor             centos-disk-large
   vim_vm_name        linux_vm
   bootup_time        -1
   recovery_wait_time 0
   recovery_policy action_on_recovery REBOOT_ONLY
   interfaces interface 0
    model   virtio
    network wan-net
   !
   scaling min_active 1
   scaling max_active 1
   placement zone_host
    host datastore1
   !
   vmexport_policy disk_exclusion Linux_IMAGE:512G-file.qcow2 =====> Disk to be excluded
on Export
   !
  !
 !
!
```

VNF Volumes

A VNF can be created and deployed with maximum two volumes. Currently, NFVIS supports a maximum of two volumes per VNF. The empty volume can be used as an extra storage by the VNF. Starting from NFVIS 4.1 release, storage volumes can now be deployed on external datastores and NFS.

Restrictions for VNF Volumes

- Volumes cannot be updated for a VNF after a VNF has already been deployed.
- NFVIS currently supports only two volumes per VNF.
- Starting from NFVIS 4.1 release, volumes can be stored in local storage or NFS, and other data stores like datastore1, datastore2 and datastore3.

Example: Payload for Creating Volumes

```
…
<volumes>
  <volume>
    <name>Volume1</name>
```

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

Volume Storage Locations

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

- <**storage_location**>*local*</**storage_location**>
- <**storage_location**>*nfs:NFS_MOUNT_NAME*</**storage_location**>
- <**storage_location**>*datastore1*</**storage_location**>
- <**storage_location**>*datastore2*</**storage_location**>
- <**storage_location**>*datastore3*</**storage_location**>

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

In NFVIS 4.1.1 release , *local* and *datastore1* are identical in storage_location.

Volume Deployment Parameters

Cisco NFVIS supports the following fields for volume deployment.

Property	Allowed Values	Description
name	string, { length 1..255 }	Name of the volume
volid	uint16	Volumes will be presented to the VM sorted by volume ID.
bus	virtio, ide, scsi	The bus type
size	unit 16	Size of the Volume
sizeunit	MiB,GiB,TiB,PiB	Size units. MiB/GiB/TiB/PiB/EiB
format	qcow2, raw	Format of the disk to be created.
device_type	disk	Type of the device being attached to the VM.
storage_location	local, datastore1, datastore2, datastore3, nfs:NFS_MOUNT_NAME	Storage location name **Note** Starting from NFVIS 4.1 release, external datastore storage location is supported.

Port Forwarding

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

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

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

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

NGIO

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

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

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

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

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

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

Requirements for NGIO Validation

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

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

VNF Deployment Using Granular RBAC

Starting from NFVIS 4.7.1, the system administrator can define a set of groups and assign VNFs to these groups. When you create a user, you can assign that user to one of these groups to control VNF access. For more information on granular RBAC and resource groups, see Granular Role-Based Access Control.

The following example shows how to create a VM with a group name:

```
nfvis(config)# vm_lifecycle tenants tenant admin deployments deployment sample resource_group
 localgroup2 vm_group test image centos flavor small vim_vm_name sample bootup_time -1
recovery_wait_time 0 recovery_policy recovery_type AUTO action_on_recovery REBOOT_ONLY
monitor_on_error false
nfvis(config-vm_group-test)# commit
```

The following example shows how to update a VM from one resource group to another:

```
nfvis(config)# vm_lifecycle tenants tenant admin deployments deployment sample
nfvis(config-deployment-sample)# no resource_group
nfvis(config-deployment-sample)# commit
Commit complete.
nfvis(config-deployment-sample)# resource_group localgroup
nfvis(config-deployment-sample)# commit
Commit complete.
```

The following example shows how to view the list of VMs with their group information:

```
nfvis# show vm_lifecycle deployments

Name          Monitored  State           Internal-State           GroupInfo
-------------------------------------------------------------------------------
centos1       No         ALIVE           VM_INERT_STATE           localgroup
cetos_global  No         ALIVE           VM_INERT_STATE
Denotes global
sample        No         ALIVE           VM_INERT_STATE           localgroup
```

The following example shows how to view the overall group, users and VM mappings:

```
nfvis# show group-summary
```

```
Group_Name              Policies                       Users                       Vms
-------------------------------------------------------------------------------------------------------
localgroup          resource-access-control        local_oper             centos1.centos1
                                                   sample.test
-------------------------------------------------------------------------------------------------------
localgroup2         resource-access-control         test_admin
-------------------------------------------------------------------------------------------------------

                           NA                           admin
cetos_global.cetos_global
                                                   localAdmin
-------------------------------------------------------------------------------------------------------
```

Deploy CUCM Application

After successfully registering the application image, use this procedure to deploy a CUCM virtual machine on Cisco NFVIS with a Bootstrap ISO for Day0 configuration.

Prerequisites

- Ensure you have a registered CUCM application image (ISO and OVA) in NFVIS.
- Ensure you have a Bootstrap ISO with Day0 configuration for the CUCM application.

  Refer to the CUCM documentation for instructions on creating the Bootstrap ISO with Day0 configuration: *Touchless Installation Task Flow*

Deploy CUCM VM via NFVIS Portal

1. From the NFVIS portal, click **Configuration** > **Deploy**
2. Select a VM type from the drop-down list.

   If your application type is not listed, select **OTHER**.
3. From the **Image** drop-down list, select the desired image.

   Images are populated based on the VM type selected. For example, `Bootable_UCSInstall_UCOS_15.0.1.iso`.
4. From the **Flavor** drop-down list, select the appropriate flavor based on your sizing requirements.

   Flavors are automatically created from the OVA during image registration.
5. Configure network connections.
   a. In the **Network Design** section, locate the VM icon on the canvas.
   b. Add additional networks as required.
   c. Drag a line from the VM icon to the desired network to connect to the appropriate network(s) for your deployment.
6. Select the **Bootstrap ISO** check box.
7. Click **Upload** and select your Day0 configuration ISO.
8. Click **Deploy**.

   The system validates the configuration. If validation errors occur, they are displayed for correction.

9. Track the deployment progress. Wait until the VM state changes to **Active** status.

   After the VM is active, click the **Terminal** button in the VM row. Monitor the installation progress via the console and follow application-specific installation prompts.

Deployment Status

This table provides details about virtual machine deployment statuses, their descriptions, and actions.

Table 39: Protocols

Status	Description	Action
DEPLOYING	VM is being created	Wait for completion
ACTIVE	VM is running	Access console, verify operation
ERROR	Deployment failed	Check logs, troubleshoot
SHUTDOWN	VM is powered off	Power on if needed

Note The deployment steps for Cisco Unity Connection, and Cisco IM and Presence Service (IM&P) on NFVIS are the same as those for Cisco Unified Communications Manager (CUCM). These steps include:

1. Image Registration – Register the ISO and OVA metadata (same process as CUCM).
2. Virtual Machine (VM) Deployment – Deploy the VM with a Bootstrap ISO (same process as CUCM).
3. Virtual Machine (VM) Management – Perform power operations, console access, and network updates (same as CUCM).

   For application-specific documentation, refer to the respective Cisco documentation for Cisco Unity Connection and Cisco IM and Presence Service.

Recovery Procedures for CUCM Applications

Recover a Cisco CUCM Application Using a Recovery ISO

Use this procedure to recover or repair a Cisco CUCM application.

1. From the NFVIS portal, click **Configuration** > **Virtual Machine** > **Images** > **Image Repository**
2. Upload the Recovery ISO.

   If the Recovery ISO registration is not allowed, upload the respective VM OVA as well and register it.

   Name the image descriptively. Example: cucm-15-recovery-iso.
3. Shut down the VM..
   a. Click **Configuration** > **Virtual Machine** > **Manage**.
   b. Select the VM.

c. Click **Switch Power** to shut down the VM.

   Wait for the VM state to move to a **SHUTDOWN** state.

4. Attach the Recovery ISO to the CD-ROM.
   a. With the VM in the **SHUTDOWN** state, click the **CD-ROM** button.
   b. Click **ATTACH**.
   c. Select the Recovery ISO from the drop-down list.
   d. Click **Submit**.

      The Recovery ISO is now attached to the VM.

5. Power on and boot from the Recovery ISO.
   a. Click **Switch Power** to start the VM.
   b. Click **Terminal** to access the console.
   c. Enter the Boot Menu by pressing the appropriate key during startup (typically F12 or ESC).
   d. Select the Recovery ISO device from the boot menu.

      The VM boots from the Recovery ISO.
   e. Follow the application-specific recovery procedure.

      This step involves interacting with the application's recovery prompts via the console.

6. Detach the CD-ROM after recovery.
   a. Shut down the VM.
   b. Click the **CD-ROM** button.
   c. Select the **DETACH** action.
   d. Select the Recovery ISO disk.
   e. Click **Submit**.

7. Power on the VM for normal operation.

Recover the Password for a Cisco CUCM Application

Use this procedure to recover or reset the password for a Cisco CUCM application while the VM is in an **ACTIVE** state. NFVIS provides the capability to eject and insert ISO media, which is necessary as part of the application's password recovery procedure.

For the complete application-specific password recovery procedure, refer to the CUCM documentation. See Reset or Change CUCM OS Admin and Security Password.

This procedure covers only the NFVIS media operations.

1. From the NFVIS portal, click **Configuration** > **Virtual Machine** > **Manage**.
2. Ensure the VM is running.

The VM must be in an **ACTIVE** state, as password recovery is performed on a running VM.

3. Eject the current ISO, if required.
   a. Select the VM.
   b. Click the **CD-ROM** button.
   c. Select **EJECT**.
   d. Select the ISO image to eject.
   e. Click **Submit**.
4. Insert the required ISO.
   a. Click the **CD-ROM** button.
   b. Select **INSERT**.
   c. Select the required ISO from the drop-down list.
   d. Select the appropriate **Device ID** for the CD-ROM.
   e. Click **Submit**.
5. Complete the password recovery.
   a. Access the VM console via the **Terminal** button.
   b. Follow the application-specific password recovery steps.

VM States

The following table describes various VM states.

VM States	Description
VM_UNDEF_STATE	The initial state of a VM or VNF before deployment of this VM.
VM_DEPLOYING_STATE	The VM or VNF is being deployed on to the NFVIS.
VM_MONITOR_UNSET_STATE	The VM or VNF is deployed in NFVIS but the monitoring rules are not applied.
VM_MONITOR_DISABLED_STATE	Due to a VM action request or recovery workflow, the monitoring or KPI rules applied to the VM were not enabled.
VM_STOPPING_STATE	VM or VNF is being stopped.
VM_SHUTOFF_STATE	VM or VNF is in stopped or shutoff state.
VM_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 States	Description
VM_ALIVE_STATE	VM or VNF is deployed and successfully booted up or alive as shown in the KPI metric.
VM_UNDEPLOYING_STATE	The deployment of a VM or VNF is being terminated.
VM_ERROR_STATE	The VM or VNF is in an error state because the deployment or some other operation has failed.

NFVIS Container Deployment

NFVIS container deployment follows the same workflow as VM deployment.

Restrictions for NFVIS Container Deployment

- Import and export of containers is not supported.

The container deployment process in container lifecycle management supports the following:

- SRIOV/Host Interface
- Day 0 Configuration
- Configuration Options
- vCPU Pinning
- Volumes

SRIOV/Host Interface

To attach an SRIOV or host interface to the container's interface, specify the SRIOV or host interface in the configuration. Either physical interfaces or SRIOV VFs can be specified in the below configuration.

To attach an SRIOV or host interface to the container's interface, use the following configuration:

```
<?xml version="1.0" encoding="UTF-8"?>
<vm_lifecycle xmlns:ns0="http://www.cisco.com/nfvis/vm_lifecycle"
xmlns="http://www.cisco.com/nfvis/vm_lifecycle">
    <tenants>
        <tenant>
            <name>admin</name>
            <deployments>
                <deployment>
                    <name>ubuntu-2</name>
                    <vm_group>
                        ...
                        <interfaces>
                            <interface>
                                <nicid>0</nicid>
                                <network>int-LAN-vf-1</network>
                            </interface>
                        </interfaces>
                    </vm_group>
                </deployment>
            </deployments>
        </tenant>
    </tenants>
</vm_lifecycle>
```

Day 0 Configuration

Day 0 support for container deployments follows the same process as that of VM deployments. Refer to VM Bootstrap Configuration Options with a VM Deployment.

Configuration Options

There are multiple boot options available, that allow you to customize the container behavior.

To customize the container behavior using different configuration options, use the following configuration:

```
<config_data>
  <config_type>CONFIG_DATA_OPTIONS</config_type> <!-- type is required or options below are
 ignored -->
  <config_options>
    <options>
      <option><name>ENV_VARIABLE</name><value>env=test</value></option>
      <option><name>ENV_VARIABLE</name><value>env2=test2</value></option>
    </options>
  </config_options>
  <configuration>
     <dst>/etc/ovf-env.xml</dst>
     . . .
```

Table 40: Supported Configuration Options

Option Name	Type	Description
ENV_VARIABLE	String	Environment variable to set within container: name/value pair (example: varName=varValue)
LOW_LATENCY	Boolean	vCPU pinning option: When the value is True, the container is allocated a dedicated core.

vCPU Pinning

Similar to the NFVIS VM model, containers can also be pinned to a specific vCPU. If a container requires a dedicated core, a low latency flag can be set using the configuration options. By default, no low latency flag is set, this means that the core may be shared with other containers. Because the workflow for container vCPU pinning is the same as VM deployments, VimManager reuses the Node.js CPU allocation API to interface with the NFVIS CPU allocation scripts.

Volumes

Volumes support for container deployments uses the same datamodel as VM deployments and provides external storage options to the containers. The `storage_location` tag in the payload refers to the file path within the container where the volume is mounted.

Note

- Volumes are equivalent to docker binds.
- All container volumes are stored on datastore1.

To mount a container using a volume or a specific path, use the following configuration:

```
<vm_group>
    <name>docker-nginx</name>
    <image>nginx</image>
    . . .
    <volumes>
      <volume>
        <name>test1</name>
        <volid>1</volid>
        <storage_location>/etc/datastore</storage_location> <!-- the path to mount inside
the container -->
        <size>120</size>
        <sizeunit>MiB</sizeunit>
      </volume>
      <!-- using default size allocations - 10 gibibytes (GiB) -->
      <volume>
        <name>test2</name>
        <volid>2</volid>
        <storage_location>/var/logs</storage_location>
      </volume>
```

For more information, see VNF Volumes.

VNF Deployment Update

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

Update VNF Flavor

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

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

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

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

Get Available Flavors

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

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

Check System CPU Usage

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

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

Update VNF Flavor

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

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

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

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

Update CPU Topology

Updating the CPU topology of a VNF involves updating a VM to a custom-defined topology that you would have created at the time of creating a VM flavor. For more details, see VM Profiles or Flavors, on page 130.

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

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

Example: Updating ISRv to Include a CPU Topology

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

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

Verify Changed Configuration

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

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

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

```
  <topology sockets='1' cores='3' threads='1'/>
</cpu>
```

About Updating VNF Interfaces

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

Hot and Cold Updates

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

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

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



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

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

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

Prerequisites for Updating VNF Interfaces

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

Supported Interface Update Operations by Interface Types

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

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

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

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

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

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

Interface update of any SRIOV interface to remove from one interface and add the same to another or swap, in a single transaction fails. Use two separate requests for a successful update.

Update Interfaces

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

Add Interfaces

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

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

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

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

```
      <interface>
      <nicid>2</nicid>
      <network>lan-net</network>
      </interface>
      </interfaces>'

HTTP/1.1 204 No Content
```

Delete Interfaces

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

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

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

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

HTTP/1.1 204 No Content
```

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

Move VNICs from One Network to Another

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

```
        <network>NewNetworkSelected</network>
    </interface>
</interfaces>'
```

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

```
curl -k -v -u admin:Esc123# -H
Accept:application/vnd.yang.data+xml -H Content-Type:application/vnd.yang.data+xml -X PUT
https://172.29.91.32/api/config/vm_lifecycle/tenants/tenant/admin/deployments/deployment/ASAdep/vm_group/ASAvmgrp/interfaces
 --data '
<interfaces>
         <interface>
                 <nicid>0</nicid>
                 <network>int-mgmt-net</network>
             </interface>
             <interface>
                 <nicid>1</nicid>
                 <network>wan2-net</network>
          </interface>
</interfaces>'

HTTP/1.1 204 No Content
```

Access VNFs

In Cisco NFVIS, you can access VNFs in three ways after they have been deployed:

- Through VNC Console
- Through Serial Console.
- Through Port Forwarding if VM is deployed with port_forwarding configuration in deployment payload

Access VNFs Using VNC Console

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

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

1. Using the NFVIS portal, from the **Manage Deployments**, click **Configuration** > **Virtual Machine** > **Manage**
2. Select the desired VM from the list. The VM must be in active status.
3. Click **Terminal** in the VM row. A console window opens in your browser.

Secure Access to the VNC Console

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

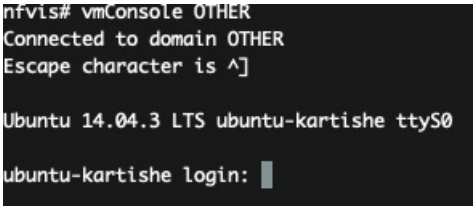
```
<vm_lifecycle>
 <tenants>
  <name>admin</name>
   <deployments>
    <deployment>
     …
```

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

Access VMs Using Serial Console

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

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



When you access the VM console using the ENCS console port on the device, you cannot use ctrl+] to exit the VM console. You must use ctrl+] and then enter **`send escape`** to exit the VM console.

```
nfvis# show system deployments
NAME                                        ID  STATE
-----------------------------------------------------
sj-02-sj02-isrv.sj-02-sj02-isrv             1   running
sj-02-sj02-linux.sj-02-sj02-linux           2   running
sj-02-sj02-vwaas.sj-02-sj02-vwaas           3   running
sj-02-sj02-firewall.sj-02-sj02-firewall     4   running

nfvis#
nfvis#
nfvis# vmConsole sj-02-sj02-vwaas.sj-02-sj02-vwaas
Connected to domain sj-02-sj02-vwaas.sj-02-sj02-vwaas
Escape character is ^]

sj-02-sj02-vwaas#
sj-02-sj02-vwaas#
sj-02-sj02-vwaas#
telnet> send escape

nfvis#
```

Access a VM Using Port Forwarding

To access a VM using port forwarding:

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

```
<port_forwarding>
  <port>
    <type>ssh</type>
```

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

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

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

isrv-encs#
```

Import and Export NFVIS VM

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

VM Export and Import Limitations

- The imported VM cannot change datastore.
- The original registered image must exist.
- The OVS network name must be identical to the one used by original deployment.
- 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
Restore backing images and flavors	Not supported	Not supported	Supported

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

Secure Boot of VNFs

A key aspect of a secure compute system is to ensure that the system is running the intended software without malware or tampered software. This protection must begin as soon as the system is powered-on. The UEFI (Unified Extensible Firmware Interface) specification defines a secure boot methodology that prevents loading software which is not signed with an acceptable digital signature.

NFVIS already supports UEFI Secure Boot of Host. Secure boot ensures that the booted NFVIS software is genuine. Starting from NFVIS 3.11.1 release, the UEFI secure boot feature has been extended to VNFs being used in the Cisco NFV solution. Starting from Cisco NFVIS release 4.18.1, NFVIS secure boot for Host is supported on Cisco UCS C-Series Rack Servers.

Booting a VNF securely requires the environment to support UEFI secure boot and requires modification to VNFs to support secure boot. NFVIS 3.11.1 and later releases have the infrastructure to support UEFI secure boot for VNFs that are capable of secure boot.



Note ISRv versions 16.12 and later support secure boot.

The firmware modes in which VNFs can boot are:

- BIOS - for VNFs which are not capable of secure boot
- UEFI secure - for VNFs capable of secure boot

VNFs can indicate secure boot capability using properties in the image_properties.xml file in the tar.gz package for the VNF.

You must set the following properties to enable secure boot of VNFs:

Name of the Image Property	Value for non-secure boot VNF	Value for secure boot VNF
boot_mode	bios	efi-secure-boot
shim_signature	N/A	microsoft **Note** For secure-boot capable VNFs, the VNF shim is typically signed by Microsoft.

Any combinations not matching the above default to the following:

- boot_mode: bios
- shim_signature: N/A



Note On the NFVIS portal, the **Image Repository** page shows if the image is capable of secure boot.

CDROM Attachment and Detachment

Information about CDROM Attachment and Detachment

CDROM attachment and detachment is a VM management capability that allows you to connect or disconnect ISO images to a VM. This feature supports various use cases, including system recovery, operating system installation, and software deployment, by enabling the VM to access content from a virtual CDROM drive.

CDROM Attachment and Detachment

Attaching a CDROM to a VM is a two-step process:

1. Image Registration: First, you must register the desired ISO image using the existing image registration workflows provided by NFVIS. This makes the ISO available for selection. For information about ISO image registration, see Image Registration.
2. CDROM Action: After registration, you can proceed to attach the registered ISO to the target VM by utilizing the dedicated CDROM actions available in the **Manage Deployments** page.

   To detach an ISO image, you would use the corresponding detachment action in the **Manage Deployments** page. Both attachment and detachment operations require the VM to be in a shutdown state.

Prerequisites for CDROM Attachment and Detachment

To successfully perform CDROM attachment or detachment operations, the VM must be in a shutdown state. These operations cannot be initiated or completed on an active VM.

Manage CDROM Attachment and Detachment

Before you begin

- Ensure the ISO image you intend to attach is registered using the image registration workflows. For more information, see Image Registration.
- Verify that the VM is in a shutdown state. CDROM attach and detach operations cannot be performed on an active VM.

Attach a CDROM to a VM

To attach an ISO image to a VM via the NFVIS portal, do the following:

1. From the **Manage Deployments** page, click the **CD-ROM** icon.
2. Click **Attach**.

3. From the **Images** drop-down list, choose the desired recovery ISO from the list of registered ISO images.
4. Click **Submit**.

   A confirmation message appears indicating the CDROM attachment is complete.

   After attaching a CDROM, to use the attached recovery ISO, you must start the VM. Then, access its boot menu to select the ISO disk for booting.

Detach a CDROM to a VM

To detach an ISO image from a VM via the NFVIS portal, do the following:

1. From the **Manage Deployments** page, click the **CD-ROM** icon.
2. Click **Detach**.
3. From the **Disk Name** drop-down list, choose the disk name corresponding to the recovery ISO you wish to detach.
4. Click **Submit**.

   A confirmation message appears indicating the CDROM detachment is complete.

   After detaching a CDROM, can start the VM for normal operation. To confirm detachment, click **CD-ROM** > **Detach**. The ISO image should no longer be listed in the **Disk Name** drop-down list.

VM Graceful Stop

Information about Graceful VM Stop

Graceful stop or graceful shutdown of VM is a power management option that sends a ACPI signal to the operating system within the VM, allowing its services and applications to terminate in an orderly manner before the VM powers off. This contrasts with a forceful shutdown, which immediately cuts power to the VM.

Key Features and Behavior

- Default Behavior: When initiating a VM shutdown, the **Graceful stop** checkbox is selected by default for VMs deployed with type as **OTHER**.
- Timeout: The graceful stop operation has a timeout period of 20 minutes.
- Error State: If the VM does not shut down gracefully within the 20-minute timeout, it will transition to an error state.
- Forceful Fallback: If a graceful stop fails or the VM enters an error state, you can uncheck the **Graceful stop** option to perform a forceful shutdown.

Perform a Graceful stop

To perform a graceful stop of a VM via the NFVIS portal:

1. From the **Manage Deployments** page, click the **Switch Power** icon for the VM that you want to stop or shutdown.

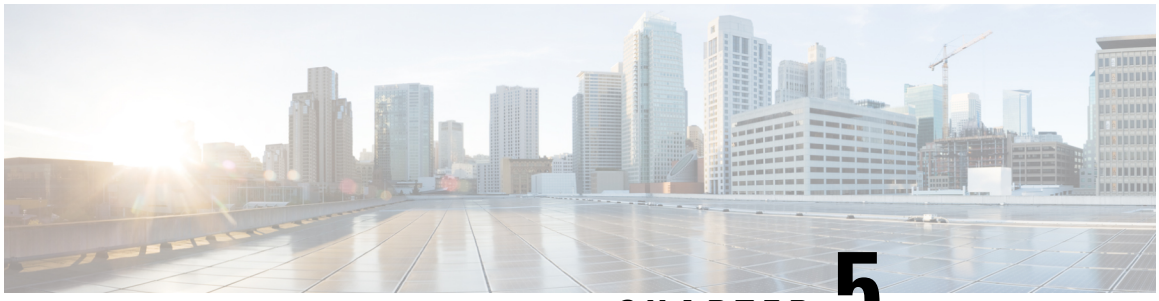
   The **Graceful stop** checkbox is selected by default for VMs deployed with type as **OTHER**.

2. Click **OK** to confirm VM shutdown.

   The graceful shutdown process begins in the background.

3. Monitor the VM status in the management interface. It should transition from **active** to **shutdown**.

   If the VM does not shut down within 20 minutes, it will automatically move to an error state. In this case, you may need to perform a forceful shutdown by repeating this procedure and unchecking the **Graceful stop** option.



CHAPTER 5

Cisco NFVIS Kubernetes Support

Table 41: Feature History

Feature Name	Release Information	Description
Cisco NFVIS Kubernetes support	Cisco NFVIS Release 4.9.1	This feature adds support to deploy and operate Kubernetes as a VM using Cisco NFVIS. Download the NFVIS Kubernetes tar.gz file, upload it to the Cisco NFVIS and deploy the Kubernetes VM to leverage Kubernetes related technologies within Cisco NFVIS.

- Information About Cisco NFVIS Kubernetes Support, on page 167
- Prerequisites for Cisco NFVIS Kubernetes Support, on page 168
- Restrictions For Cisco NFVIS Kubernetes Support, on page 168
- Deploy Kubernetes on Cisco NFVIS, on page 168
- Verify Cisco NFVIS Kubernetes Deployment, on page 170
- Verify Kubernetes Deployment using CLI, on page 171
- Access Kubernetes Cluster, on page 172
- Troubleshoot Cisco NFVIS Kubernetes VM Support, on page 173

Information About Cisco NFVIS Kubernetes Support

Information About Cisco NFVIS Kubernetes Support

Starting from Cisco NFVIS Release 4.9.1, Cisco NFVIS leverages Kubernetes capabilites to enable cloud native deployments. The Cisco NFVIS Kubernetes support enables deploying edge compute applications on the branch and edge devices. Cisco NFVIS can handle the Life Cycle Management of Kubernetes and enable service chaining with other VNFs. Cisco NFVIS contains built-in support for creating Kubernetes topology with Cisco Catalyst 8000V.

Information About Kubernetes as a VM

Kubernetes is packaged as a Virtual Machine (VM) that is deployed on Cisco NFVIS.

Prerequisites for Cisco NFVIS Kubernetes Support

Ensure that the minimum software version for Cisco NFVIS devices is Cisco NFVIS Release 4.9.1.

Restrictions For Cisco NFVIS Kubernetes Support

- Cisco NFVIS supports only single node topology. A single node contains both the control and data planes.
- Only one Kubernetes node can be deployed at a time on Cisco NFVIS.

Deploy Kubernetes on Cisco NFVIS

Kubernetes VM can be deployed using Cisco NFVIS. Cisco NFVIS supports deploying of Kubernetes VM using the Cisco NFVIS Portal, API, and CLI.

Download the latest Kubernetes tar.gz image file from the Cisco Software download page on to your local storage for version mapping with Cisco NFVIS Release 4.9.1.

Upload the Kubernetes tar.gz image file to the Cisco NFVIS and register the image. For more information on uploading and registering the image files see, Uploading VM Images to an NFVIS Server and Registering a VM image.

Deploy Cisco Kubernetes Using Cisco NFVIS

Deploy Cisco NFVIS Kubernetes VM

Use the following steps to deploy Kubernetes VM:

1. In Cisco NFVIS, click **Configuration** > **Deploy**.
2. In the **Select VM** menu, click **KUBERNETES** and the Kubernetes node is displayed inside the topology.



Note A **kubernetes-net** network is auto-created in the topology when you click **KUBERNETES**.

We recommend that you establish a connection between the Kubernetes and WAN-Net that can access the APISERVER_EXTERNAL_IP to complete the Kubernetes deployment with your existing network using the topology.

3. Connect the Kubernetes to a preferred network interface using the topology.



Note Kubernetes VM can be connected to a maximum of two network interfaces. The vNIC ID of the first interface (management-interface) is zero and the vNIC ID of the second interface (data-path) is one.

In the **vNIC Details** panel, the **VM Name** and **Network Name** are auto-populated.

4. Click **Kubernetes** inside the topology to expand the **Kubernetes deployment** panel.
5. Configure the following fields:

Field	Description
VM Name	Enter a VM name.
Image	Choose an image from the drop-down or click + button to add a new image.
Profile	Choose the profile for the VM from the drop-down or click the + button to add a new profile. **Note** Cisco NFVIS supports only small and medium profiles for Kubernetes deployment.
Group Name (optional)	Choose a group name from the drop-down list.
VNC Password	Enter the VNC password.
MODE	Choose **control-plane-node** from the drop-down.
GATEWAY	Enter the Cisco NFVIS gateway IP address.
NODE_IP_ADDRESS	Enter the node IP address. This IP address is assigned to the interface on vNIC ID one.
HOSTNAME	Enter a host name.
APISERVER_EXTERNAL_IP	Enter the API server IP address of an external node.
NTP_SERVER_IP	Enter the NTP server IP address to configure an NTP server to the Kubernetes VM.
Deployment Disk	Choose a deployment disk from the drop-down. **Note** Do not choose NFS Store as a deployment disk from the drop-down.
Management IP (optional)	Enter the IP address of the management interface on vNIC ID zero. If left blank, NFVIS auto-populates this field.

6. Click **Deploy**.

Deploy Kubernetes As a Service Chain using Cisco NFVIS

Deploy Kubernetes in service chains using the Kubernetes Topology method which creates a new topology with an established kubernetes-net network. Use the following steps to deploy Kubernetes as a service chain:

Note

- Ensure that a Cisco Catalyst 8000V gateway is uploaded to NFVIS in the image repository.
- You can deploy Kubernetes as a service chain only using Cisco NFVIS.
- The Kubernetes topology deployment deploys both Kubernetes and Cisco Catalyst 8000V VMs.

1. Perfom the steps 1 to 5 mentioned in the Deploy Kubernetes as a standalone VM topic.

Note Enter the Cisco Catayst 8000V gateway IP address in the **GATEWAY** filed, enter the **NODE_IP_ADDRESS** and **APISERVER_External_IP** of Cisco Catalyst 8000V.

2. Ensure to check the **Kubernetes Topology** checkbox.
3. Configure the following fields:

Field	Description
Image Name	Choose an image from the drop-down.
Profile Name	Choose the profile name from the drop-down.
VNC password	Enter the VNC password.
C8KV_WAN_IP	Enter the WAN IP address of Cisco Catayst 8000V.
C8KV_WAN_GATEWAY	Enter the WAN gateway of Cisco Catalyst 8000V.
Tech Package	Choose the appropirate package from the drop-down.
SSH_USERNAME	Enter the SSH username.
SSH_PASSWORD	Enter the SSH password.

4. Click **Deploy**.

Verify Cisco NFVIS Kubernetes Deployment

To verify the status of the Kubernetes deployment using the Cisco Enterprise NFVIS, use the following steps:

1. In the Cisco Enterprise NFVIS dashboard, click **Configuration** > **Virutal Machine** > **Manage**.
2. In the **Manage VM** page, the deployed Kubernetes VM is listed.

Verify Kubernetes Deployment using CLI

Verify Kubernetes Status

Once the Kubernetes deployment is complete, verify the status of the Kubernetes cluster. The following is a sample output from the **kubernetes status** command:

```
nfvis# kubernetes status
nfvis# kubernetes status
NAME       STATUS   ROLES                  AGE   VERSION
hostname   Ready    control-plane,master   55s   v1.22.4
```

In this output, **Ready** indicates that the cluster is active.

Generate Configuration to access the Kubernetes Cluster

To get the configuration and credentials to access a Kubernetes cluster using Cisco NFVIS use the following command:

```
nfvis# kubernetes getConfig
kubernetes-config apiVersion: v1
clusters:
- cluster:
    certificate-authority-data: xyxyajsuus
    server: https://209.165.201.1:0001
  name: kubernetes
contexts:
- context:
    cluster: kubernetes
    namespace: nfvis-kubernetes
    user: nfvis-kubernetes-admin
  name: nfvis-kubernetes-admin
current-context: nfvis-kubernetes-admin
kind: Config
users:
- name: nfvis-kubernetes-admin
  user:
    token: xxxyyzha
```

In this output, you can get the configuration details such as the token number, name of the cluster, user name and certification details.

Regenerate Updated Kubernetes Configuration

To get the updated service token details and Kubernetes configuration from the cluster, use the following command:

```
nfvis# kubernetes regenerateConfig
kubernetes-config apiVersion: v1
clusters:
- cluster:
    certificate-authority-data: XXYZZZUS
    server: https://209.165.201.1:0001
  name: kubernetes
contexts:
- context:
    cluster: kubernetes
    namespace: nfvis-kubernetes
    user: nfvis-kubernetes-admin
  name: nfvis-kubernetes-admin
```

```
current-context: nfvis-kubernetes-admin
kind: Config
users:
- name: nfvis-kubernetes-admin
  user:
    token: Yl4RMEg
```

In this output, you can get the latest configuration details such as the token number, name of the cluster, user name and certification details.

Access Kubernetes Cluster

Information about Kubernetes Dashboard

- The Kubernetes Dashboard is hosted on the Kubernetes cluster and provides a GUI to access the cluster.
- The Kubernetes cluster can only be accessed via the Kubernetes Dashboard or API.

Note A Kubernetes node needs to be deployed in Cisco NFVIS and should be in the **VM_ALIVE** state.

- The dashboard is connected to Kubernetes over port 32045. For example, https://API_SERVER_EXTERNAL_IP:32045

Access Kubernetes Cluster using Cisco NFVIS

There are two options to access a Kubernetes cluster using Cisco NFVIS. The following sections brief about accessing the Kubernetes dashboard using **Kubernetes Config** and **Kubernetes Token** options.

Access Kubernetes Dashboard using Kubernetes Config Option

Use the following steps to access the Kubernetes dashboard:

1. In the Cisco Enterprise NFVIS dashboard, click **Configuration** > **Virutal Machine** > **Manage**.
2. In the **Manage VM** page, the deployed Kubernetes VMs are listed.
3. Click **Kubernetes Config** icon next to the respective Kubernetes VM.
4. Kubernetes Config page is displayed. Make a note of the server IP address displayed in the configuration.
5. Click **Copy to clipboard** to copy the configuration to your clipboard in case of using the **Kubernetes Config** option.

Note You can get the **Kubernetes Config** using the CLI as well. Refer to the verify Kubernetes cluster using CLI topic for more information.

6. Open a new tab in your preferred browser and enter the API server IP address followed by the port number. For example, https://198.51.100.253:32045.

7. The Kubernetes gateway login page is displayed. Paste the entire copied configuration.
8. Kubernetes dashboard is displayed. In the dashboard, navigate to the **Namespace** called **nfvis-kubernetes**.
9. You can start using Kubernetes related technologies using the Kubernetes dashboard.

Access Kubernetes Dashboard using Kubernetes Token Option

Use the following steps to access the Kubernetes dashboard:

1. In the Cisco Enterprise NFVIS dashboard, click **Configuration** > **Virutal Machine** > **Manage**.
2. In the **Manage VM** page, the deployed Kubernetes VMs are listed.
3. Click **Kubernetes Config** icon next to the respective Kubernetes VM.
4. Kubernetes Config page is displayed. Make a note of the API server IP address displayed in the configuration.
5. Select the token details and copy it to your clipboard in case of using the **Token** option to access the dashboard.
6. Open a new tab in your preferred browser and enter the API server IP address followed by the port number. For example, https://198.51.100.253:32045.
7. The Kubernetes gateway login page is displayed. Paste the copied token details in the **Using Token** option.



Note

Cisco NFVIS Kubernetes- Parameters	Value
Namespace	nfvis-kubernetes
Username	nfvis-kubernetes-admin
Dashboard Port	32045
Ingress Controller Port	32050

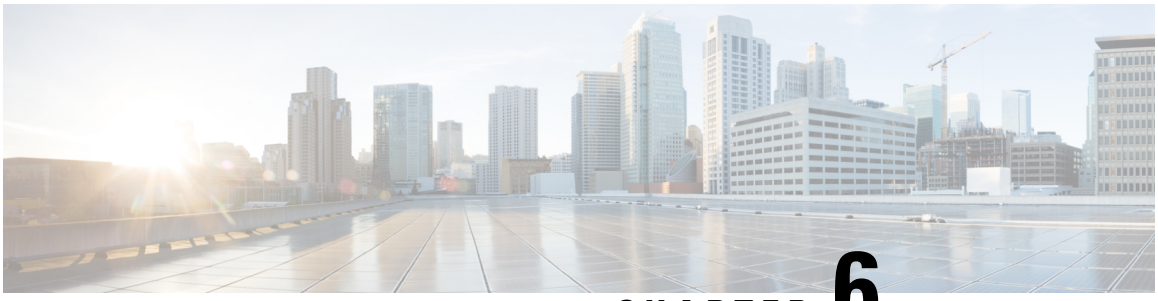
8. Kubernetes dashboard is displayed. In the dashboard, navigate to the **Namespace** called **nfvis-kubernetes** by searching for it.
9. You can start using Kubernetes related technologies using the Kubernetes dashboard.

Troubleshoot Cisco NFVIS Kubernetes VM Support

Log Files

The following commands are used to get the logs for troubleshooting:

Command	Description
`show log kubernetes`	Displays a list of available logs specific to the Kubernetes VM.
`show log kubernetes boot.log`	Displays logs containing Kubernetes VM boot details.
`show log kubernetes journal`	Displays logs containing Kubernetes VM journal binary.
`show log kubernetes cloud-init.log`	Displays logs containing Kubernetes VM cloud init logs.
`show log kubernetes cloud-init-output.log`	Displays logs containing Kubernetes VM cloud init output.
`show log kubernetes KubeHelper.log`	Displays logs containing Kubernetes VM helper service output.
`show log kubernetes KubeHelperExceptions.log`	Displays logs containing Kubernetes VM helper service exceptions.
`show log kubermetes messages`	Displays logs containing Kubernetes VM system service output.



CHAPTER 6

Cisco NFVIS ThousandEyes Support

Table 42: Feature History

Feature Name	Release Information	Description
Cisco NFVIS ThousandEyes Support	Cisco NFVIS Release 4.15.2	This feature adds support to deploy and operate Cisco ThousandEyes as a container using Cisco NFVIS. Download the NFVIS ThousandEyes agent container image from ThousandEyes, and deploy the ThousandEyes container to leverage ThousandEyes related technologies within Cisco NFVIS. This feature also adds support for ThousandEyes Docker Container Upgrade using the NFVIS VM Lifecycle Workflow. It also introduces support for using new NFVIS-specific Docker images provided by ThousandEyes.
Cisco NFVIS ThousandEyes Support	Cisco NFVIS Release 4.15.3	This feature allows upgrading the NFVIS ThousandEyes agent container image directly from the Cisco NFVIS web portal. You can choose from registered images or upgrade to new images.

- Information About Cisco NFVIS ThousandEyes Support, on page 176
- Prerequisities For Cisco NFVIS ThousandEyes Support, on page 176
- Restrictions For Cisco NFVIS ThousandEyes Support, on page 176
- Deploy ThousandEyes on Cisco NFVIS, on page 176
- Access ThousandEyes Using Cisco NFVIS, on page 186

Information About Cisco NFVIS ThousandEyes Support

Starting from Cisco NFVIS Release 4.15.2, Cisco NFVIS supports Cisco ThousandEyes to provide a pre-integrated solution that allows deploying ThousandEyes Enterprise Agents as a container on Cisco NFVIS. This enables running ThousandEyes network monitoring and testing capabilities directly on your Cisco NFVIS infrastructure. The feature provides visibility into the performance of the underlying network infrastructure.

Cisco NFVIS also supports ThousandEyes Docker Container Upgrade using existing VM lifecycle workflow. The VM lifecycle allows seamless image updates on the existing ThousandEyes Container deployment. Additionally, ThousandEyes has released NFVIS specific Docker container images with install type as NFVIS. You must specify the nfvis.docker ThousandEyes container image download URL in the update deployment request. NFVIS will then download the image and complete the upgrade of existing container to the specified image.

Benefits of Cisco NFVIS ThousandEyes Support

- Gain end-to-end visibility into the performance of your network infrastructure, including cloud providers, WAN links, and internal data center networks within Cisco NFVIS providing end-to-end visibility into network performance.
- Reduce mean time to resolution (MTTR) for network issues, improve network reliability, and optimize application performance.

Prerequisities For Cisco NFVIS ThousandEyes Support

- Ensure that the minimum software version for Cisco NFVIS devices is Cisco NFVIS Release 4.15.2.
- Ensure that your devices are meeting the minimum hardware requirements. For more information on the minimum hardware requirements see, Enterprise Agent System Requirements in the ThousandEyes documentation.
- Configure nameservers on your Cisco NFVIS devices to enable the download of the ThousandEyes Docker image from downloads.thousandeyes.com.

Restrictions For Cisco NFVIS ThousandEyes Support

- Starting with Cisco NFVIS Release 4.15.2, use the ThousandEyes Enterprise Agent Docker images recommended by the ThousandEyes dashboard install page.
- Deployment of ThousandEyes on Cisco NFVIS is only supported as a Docker container, not as a Virtual Machine .

Deploy ThousandEyes on Cisco NFVIS

ThousandEyes can be installed using the NFVIS web interface. Alternatively, you can use the command-line option found in the ThousandEyes documentation for installation.

Deploy ThousandEyes Container from NFVIS Web Interface

Download the latest **ThousandEyes** container image file from **Add New Enterprise Agent** dialog.

Upload the **ThousandEyes** container image file to the Cisco NFVIS and register the image. For more information on uploading and registering the image files see, Uploading VM Images to an NFVIS Server and Registering NFVIS Container Image.

Use the following steps to deploy ThousandEyes VM:

1. In Cisco NFVIS portal, click **Configuration** > **Deploy**.
2. In the **Select VM or Container** menu, click **TE** and the ThousandEyes node is displayed inside the topology.
3. Configure the following fields in the **VM Details** window.

Field	Description
VM Name	Enter the container's name.
Image	Choose the registered **thousandeyes.docker** image from the drop-down or click + button to add a new image.
Profile	Choose the profile for the ThousandEyes agent from the drop-down or click the + button to add a new profile. For more information, see Enterprise Agent System Requirements for agent without browser-bot version.
Group Name	(Optional) Choose a group name from the drop-down list.
Deployment Disk	Choose a deployment disk from the drop-down. **Note** Do not choose NFS Store as a deployment disk from the drop-down.
Add Bootstrap Config	Add bootstrap config data in the explanded **Add Bootstrap Config** pane. Check the **Is External File Path** check box if your bootstrap file path is from an external source.
Add Config Options	Add the additional configuration in the expanded **Add Config Options** pane. Choose the **Config Option Name**, Enter the **Config Option Value** after = in the pre-populated config option value. For example, TEAGENT_INET=4
Add Volume	Volume required to enable ThousandEyes on Cisco NFVIS are pre-populated with the minimum required volume. You can add or remove volumes based on your requirement.

Field	Description
Remove Volume	You can remove volumes based on your requirement.
Exclude Disks from Export	Not supported by ThousandEyes.

4. Click **Deploy**.

Upgrade ThousandEyes Agent Container from NFVIS Web Interface

Starting from Cisco NFVIS Release 4.15.3, you can upgrade the NFVIS ThousandEyes agent container image directly from the Cisco NFVIS web portal. Use the **Upgrade** icon to perform the upgrade from the **Manage Deployments** page.

Registered Image

Follow these steps to upgrade a registered image using the NFVIS web interface:

1. From Cisco NFVIS portal, click **Configuration** > **Virtual Machine** > **Manage**.
2. Click the **Upgrade** icon.
3. Choose the image from a drop-down list, displays all previously registered images.
4. Click **Submit** to upgrade your existing configuration to the selected image.

New Image

Follow these steps to upgrade using a new image from the NFVIS web interface:

1. From Cisco NFVIS portal, click **Configuration** > **Virtual Machine** > **Manage**.
2. Click the **Upgrade** icon.
3. Click **New Image**.
4. Choose a protocol.

Field	Description
HTTPS	The server path is prefilled. Enter the image name.
Docker	The repository name is prefilled. Enter the tag name of the Docker image.

5. Click **Submit** to register your new image.

Deploy ThousandEyes Container on NFVIS Using CLI Configs

Execute the following steps to deploy ThousandEyes Container on NFVIS using the CLI:

1. Register the ThousandEyes Docker Image

Note The URL in **src** is provided by **Add New Enterprise Agent dialog**.

```
vm_lifecycle images image thousandeyes-enterprise-agent-0.33.0
 src
https://downloads.thousandeyes.com/enterprise-agent/thousandeyes-enterprise-agent-0.33.0-nfvis.docker

 locator vim_id container
 properties property vnf_type
  value THOUSANDEYES
 !
!
```

2. Configure Resource Details as a Flavor

```
vm_lifecycle flavors flavor thousandeyes-flavor vcpus 2 memory_mb 1024 root_disk_mb 20480
!
```

3. Deploy the ThousandEyes Agent Container

```
vm_lifecycle tenants tenant admin
 deployments deployment TE_DEMO
  vm_group TE_DEMO
   vim_vm_name  TE_DEMO
   locator vim_id container
   image       thousandeyes-enterprise-agent-0.33.0
   flavor      thousandeyes-flavor
   bootup_time -1
   config_data configuration bootstrap_config
    data            "{ \"env_variables\" : { \"TEAGENT_ACCOUNT_TOKEN\" :
\"${TEAGENT_ACCOUNT_TOKEN}\", \"TEAGENT_INET\" : \"${TEAGENT_INET}\"} }"
    template_engine VELOCITY
    variable TEAGENT_ACCOUNT_TOKEN
     val [ 53rettywagbuouhw06hu65767rtyuyyui ]
    !
    variable TEAGENT_INET
     val [ 4 ]
   !
  !
 !
```

4. Commit the configuration.

Upgrade ThousandEyes Agent Container Using the CLI

To upgrade the ThousandEyes agent container using CLI configurations, you have two main options:

Option 1: Register and Update the Image

1. Register the New Docker Image

   Use the following command to register the newer version of the ThousandEyes agent Docker image:

```
vm_lifecycle images image thousandeyes-enterprise-agent-0.34.0
 src
https://downloads.thousandeyes.com/enterprise-agent/thousandeyes-enterprise-agent-0.34.0-nfvis.docker

 locator vim_id container
 properties property vnf_type
  value THOUSANDEYES
 !
!
```

2. Update the Container Deployment Configuration

   Update the existing container deployment to use the newly registered image:

```
vm_lifecycle tenants tenant admin
 deployments deployment TE_DEMO
  vm_group TE_DEMO
   vim_vm_name  TE_DEMO
   locator vim_id container
   image       thousandeyes-enterprise-agent-0.34.0
 !
```

3. Commit the configuration.

   NFVIS will handle downloading the image and upgrading the container.

Option 2: Direct Image URL Update

You can skip the image registration step by directly providing the ThousandEyes agent container image URL. NFVIS will automatically download the image, register it with Docker, and upgrade the container to the specified image.

```
vm_lifecycle tenants tenant admin
 deployments deployment TE_DEMO
  vm_group TE_DEMO
   vim_vm_name  TE_DEMO
   locator vim_id container
   image
https://downloads.thousandeyes.com/enterprise-agent/thousandeyes-enterprise-agent-0.34.0-nfvis.docker

 !
```

Proxy Configuration

If your NFVIS is behind a proxy, ensure to configure these proxy settings. For more information, see Configuring an Enterprise Agent to Use a Proxy Server for other ThousandEyes Agent environment variables.

1. Image Registration with Proxy

```
vm_lifecycle images image thousandeyes-enterprise-agent-0.34.0
 src
https://downloads.thousandeyes.com/enterprise-agent/thousandeyes-enterprise-agent-0.34.0-nfvis.docker

 locator vim_id container
 properties property vnf_type
  value THOUSANDEYES
 !
 properties property http_proxy
  value [ http://proxy.com ]
 !
 properties property https_proxy
  value [ http://proxy.com ]
 !
 properties property no_proxy
  value [ .cisco.com,10.1.1.1 ]
 !
!
```

2. Deployment Configuration with Proxy

```
vm_lifecycle tenants tenant admin
 deployments deployment TE_DEMO
  vm_group TE_DEMO
   vim_vm_name  TE_DEMO
```

```
   locator vim_id container
   image       thousandeyes-enterprise-agent-0.33.0
   flavor      thousandeyes-flavor1
   bootup_time -1
   config_data configuration bootstrap_config
    data            "{ \"env_variables\" : { \"TEAGENT_ACCOUNT_TOKEN\" :
\"${TEAGENT_ACCOUNT_TOKEN}\", \"TEAGENT_INET\" : \"${TEAGENT_INET}\", \"http_proxy\" :
\"${http_proxy}\", \"https_proxy\" : \"${https_proxy}\", \"no_proxy\" : \"${no_proxy}\"}
 }"
    template_engine VELOCITY
    variable TEAGENT_ACCOUNT_TOKEN
     val [ your token goes here ]
    !
    variable TEAGENT_INET
     val [ 4 ]
    !
    variable http_proxy
     val [ http://proxy.com:80/ ]
    !
    variable https_proxy
     val [ http://proxy.com:80/ ]
    !
    variable no_proxy
     val [ .cisco.com,10.1.1.1 ]
    !
  !
 !
```

Deploy ThousandEyes Container on NFVIS Using NETCONF

These NETCONF payloads are examples that can be sent to NFVIS via a NETCONF client to manage the lifecycle of the Cisco ThousandEyes Container



Note Ensure you merge these payloads into the existing configuration to avoid deleting other configurations. Refer to your NETCONF client documentation for merging procedures.

Image Registration

1. From Docker Hub

```
<?xml version="1.0"?>
<vm_lifecycle xmlns="http://www.cisco.com/nfvis/vm_lifecycle">
  <images>
    <image>
      <name>te_latest</name>
      <src>docker://thousandeyes/enterprise-agent:latest-agent</src>
      <credentials>
        <username>${DOCKER_USERNAME_HERE}</username> <!--Credentials are optional-->
        <password>${DOCKER_PAT}</password>
      </credentials>
      <properties>
        <property>
         <name>http_proxy</name> <!--Proxies are optional and depend on your topology-->

          <value>http://example.com:80/</value>
        </property>
        <property>
          <name>https_proxy</name>
          <value>http://example.com:80/</value>
        </property>
```

```
          <property>
            <name>vnf_type</name>
            <value>THOUSANDEYES</value>
          </property>
        </properties>
        <locator>
          <vim_id>container</vim_id>
        </locator>
      </image>
    </images>
  </vm_lifecycle>
```

2. From ThousandEyes Webserver

```
<?xml version="1.0"?>
<vm_lifecycle xmlns="http://www.cisco.com/nfvis/vm_lifecycle">
  <images>
    <image>
      <name>te_latest</name>

<src>https://downloads.thousandeyes.com/enterprise-agent/thousandeyes-enterprise-agent-0.34.0-nfvis.docker</src>

      <properties>
        <property>
         <name>http_proxy</name> <!--Proxies are optional and depend on your topology-->

          <value>http://example.com:80/</value>
        </property>
        <property>
          <name>https_proxy</name>
          <value>http://example.com:80/</value>
        </property>
        <property>
          <name>vnf_type</name>
          <value>THOUSANDEYES</value>
        </property>
      </properties>
      <locator>
        <vim_id>container</vim_id>
      </locator>
    </image>
  </images>
</vm_lifecycle>
```

Flavor Registration

```
<?xml version="1.0"?>
<vm_lifecycle xmlns="http://www.cisco.com/nfvis/vm_lifecycle">
  <flavors>
    <flavor>
      <name>TE</name>
      <vcpus>2</vcpus>
      <memory_mb>1024</memory_mb>
      <root_disk_mb>8192</root_disk_mb>
    </flavor>
  </flavors>
</vm_lifecycle>
```

Deployment

```
<?xml version="1.0" encoding="UTF-8"?>
<vm_lifecycle xmlns="http://www.cisco.com/nfvis/vm_lifecycle">
  <tenants>
    <tenant>
      <name>admin</name>
```

```
      <deployments>
        <deployment>
          <name>te_nfvis</name>
          <vm_group>
            <name>te_nfvis</name>
            <image>te_latest</image>
            <flavor>TE</flavor>
            <bootup_time>-1</bootup_time>
            <config_data>
              <configuration>
                <dst>bootstrap_config</dst>
                <data>{
    "env_variables" : {
        "TEAGENT_ACCOUNT_TOKEN" : "${TEAGENT_ACCOUNT_TOKEN}",
        "TEAGENT_INET" : "${TEAGENT_INET}",
        "http_proxy" : "${http_proxy}",
        "https_proxy" : "${https_proxy}",
        "no_proxy" : "${no_proxy}"
    }
}</data>
                 <variable>
                    <name>TEAGENT_INET</name>
                    <val>4</val>
                </variable>
                <variable>
                    <name>TEAGENT_ACCOUNT_TOKEN</name>
                    <val>53rettywagbuouhw06hu65767rtyuyyui</val>
                </variable>
                <variable>
                    <name>https_proxy</name>
                    <val>http://example.com:80/</val>
                </variable>
                <variable>
                    <name>http_proxy</name>
                    <val>http://example.com:80/</val>
                </variable>
                <variable>
                    <name>no_proxy</name>
                    <val>.example.com,1.2.3.4</val>
                </variable>
              </configuration>
            </config_data>
          </vm_group>
        </deployment>
      </deployments>
    </tenant>
  </tenants>
</vm_lifecycle>
```

ThousandEyes Container Deployment Upgrade on NFVIS Using NETCONF

```
<?xml version="1.0" encoding="UTF-8"?>
<vm_lifecycle xmlns="http://www.cisco.com/nfvis/vm_lifecycle">
  <tenants>
    <tenant>
      <name>admin</name>
      <deployments>
        <deployment>
          <name>te_nfvis</name> <!--The deployment name and vm_group name MUST match the
existing deployment that you wish to upgrade-->
          <vm_group>
            <name>te_nfvis</name>

<image>https://downloads.thousandeyes.com/enterprise-agent/thousandeyes-enterprise-agent-0.35.0-nfvis.docker</image>
```

```
 <!--Only image tag needs to be updated with URL or docker tag pointing to the new TE
image-->
          </vm_group>
        </deployment>
      </deployments>
    </tenant>
  </tenants>
</vm_lifecycle>
```

ThousandEyes Container Deployment on NFVIS Using RESTCONF

The following are sample RESTCONF payloads that can be sent to NFVIS using `curl` or an equivalent client to manage the lifecycle of the Cisco ThousandEyes Container:

Image Registration

1. From Docker Hub

```
curl -k -v -u admin:Cisco123# -X POST
'https://172.29.91.33/restconf/data/vmlc:vm_lifecycle/images' \
--header 'Content-Type: application/yang-data+xml' \
--data '<image>
        <name>thousandeyes-enterprise-agent-0.34.0</name>
        <src>docker://thousandeyes/enterprise-agent:0.34.0-agent</src>
    <locator>
        <vim_id>container</vim_id>
    </locator>
    <properties>
        <property>
            <name>vnf_type</name>
            <value>THOUSANDEYES</value>
        </property>
        <property>
            <name>https_proxy</name> <!--Proxies are optional and depend on your
topology-->
            <value>http://proxy.com:80/</value>
        </property>
        <property>
            <name>no_proxy</name>
            <value>.cisco.com,10.1.1.1</value>
        </property>
    </properties>
</image>'
```

2. From ThousandEyes Webserver

```
curl -k -v -u admin:password -X POST
'https://nfvis_host_ip/restconf/data/vmlc:vm_lifecycle/images' \
--header 'Content-Type: application/yang-data+xml' \
--data '<image>
        <name>thousandeyes-enterprise-agent-0.34.0</name>

<src>https://downloads.thousandeyes.com/enterprise-agent/thousandeyes-enterprise-agent-0.33.0-nfvis.docker</src>

    <locator>
        <vim_id>container</vim_id>
    </locator>
    <properties>
        <property>
            <name>vnf_type</name>
            <value>THOUSANDEYES</value>
        </property>
        <property>
            <name>https_proxy</name> <!--Proxies are optional and depend on your
```

```
    topology-->
                <value>http://proxy.com:80/</value>
            </property>
            <property>
                <name>no_proxy</name>
                <value>.cisco.com,10.1.1.1</value>
            </property>
        </properties>
    </image>'
```

Flavor Creation

```
curl -k -v -u admin:password -X POST
'https://nfvis_host_ip/restconf/data/vmlc:vm_lifecycle/flavors' \
--header 'Content-Type: application/yang-data+xml' \
--data '<flavor>
    <name>thousandeyes-flavor</name>
    <vcpus>2</vcpus>
    <memory_mb>1024</memory_mb>
    <root_disk_mb>20480</root_disk_mb>
</flavor>'
```

ThousandEyes Agent Container Deployment

```
curl -k -v -u admin:password -X POST
'https://<nfvis_host_ip>/restconf/data/vmlc:vm_lifecycle/tenants/tenant=admin/deployments'
 \
--header 'Content-Type: application/yang-data+xml' \
--data '<deployment>
          <name>TE_DEMO</name>
          <vm_group>
            <name>TE_DEMO</name>
            <vim_vm_name>TE_DEMO</vim_vm_name>
            <locator>
              <vim_id>container</vim_id>
            </locator>
            <image>thousandeyes-enterprise-agent-0.33.0</image>
            <flavor>thousandeyes-flavor</flavor>
            <bootup_time>-1</bootup_time>
            <config_data>
              <configuration>
              <dst>bootstrap_config</dst>
              <data>{ "env_variables" : { "TEAGENT_ACCOUNT_TOKEN" :
"${TEAGENT_ACCOUNT_TOKEN}", "TEAGENT_INET" : "${TEAGENT_INET}" } }</data>
              <template_engine>VELOCITY</template_engine>
              <variable>
                <name>TEAGENT_ACCOUNT_TOKEN</name>
                <val>53rettywagbuouhw06hu65767rtyuyyui</val>
              </variable>
              <variable>
                <name>TEAGENT_INET</name>
                <val>4</val>
              </variable>
              </configuration>
            </config_data>
          </vm_group>
        </deployment>
     '
```

ThousandEyes Container Deployment Upgrade on NFVIS Using RESTCONF

To upgrade the existing ThousandEyes container deployment with a newer container image, you need to update the deployment configuration with the new image details. Here's a sample RESTCONF command using a Docker Hub image:

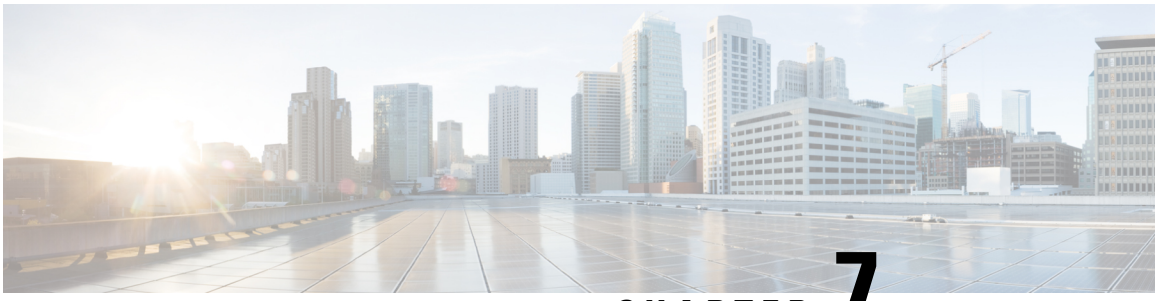
```
curl -k -v -u admin:password -X PUT
'https://<nfvis_host_ip>/restconf/data/vmlc:vm_lifecycle/tenants/tenant=admin/deployments/deployment=TE_DEMO/vm_group=TE_DEMO/image'
 \
--header 'Content-Type: application/yang-data+xml' \
--data '<image>docker://thousandeyes/enterprise-agent:0.34.0-agent</image>'
```



Note If an upgrade fails due to issues such as an image download error, verification failure, or any other problem, the container will continue running with the old image. However, the NFVIS configuration will reflect the latest settings. This behavior is similar to how VM deployment updates are handled.

Access ThousandEyes Using Cisco NFVIS

ThousandEyes **Enterprise Agents** displays the status of the Enterprise Agents deployed in Cisco NFVIS. For more information see, ThousandEyes Documentation.



CHAPTER 7

Cisco NFVIS Support For Cisco UCS C-Series Rack Servers

Table 43: Feature History

Feature Name	Release Information	Description
Cisco NFVIS Support For Cisco UCS C-Series M6 Rack Servers	Cisco NFVIS Release 4.13.1	Starting from Cisco NFVIS Release 4.13.1, Cisco NFVIS software is supported on Cisco UCS C-Series M6 Rack Servers as a term-based license. Order Cisco NFVIS software smart license along with the Cisco UCS C-Series M6 Rack Servers product purchase.
Cisco NFVIS Support For Cisco UCS C-Series M5 and M7 Rack Servers	Cisco NFVIS Release 4.18.1	Starting from Cisco NFVIS Release 4.18.1, Cisco NFVIS software is supported on Cisco UCS C-Series M5 and M7 Rack Servers with a term-based license. Order the Cisco NFVIS software smart license along with the Cisco UCS C-Series M5 and M7 Rack Servers product purchase.

- Information About Cisco NFVIS Support, on page 188
- Supported Devices For Cisco NFVIS Support For Cisco UCS Rack Servers, on page 188
- Prerequisites For Cisco NFVIS Support For Cisco UCS C-Series Rack Servers, on page 189
- Restrictions For Cisco NFVIS Support For Cisco UCS C-Series Rack Servers, on page 189
- Configure Cisco Smart Licensing for Cisco NFVIS, on page 189
- Monitor Cisco NFVIS Support for Cisco UCS C-Series Rack Server, on page 191
- Action Commands, on page 194

Information About Cisco NFVIS Support

Cisco Smart Licensing for Cisco NFVIS on UCSC Series Rack Servers

Cisco Smart Licensing is a flexible and unified licensing model that centralizes the management of all Cisco software licenses across organizations. Cisco Smart Licensing enables you to manage all product licenses from one central location. The Cisco NFVIS smart licenses are not node-locked to the device, so they can be used across all devices in your organization. The Smart Licenses can be easily transferred between devices, without hosting or issuing the license again, or going through the Return Material Authorization (RMA), or license re-issuance. For more information on Smart Licenses see, Cisco Software Licensing Guide.

Starting with Cisco NFVIS Release 4.13.1, the software supports Cisco UCS C-Series M6 Rack Servers with a term-based license. Beginning with Cisco NFVIS Release 4.18.1, this support extends to Cisco UCS C-Series M5 and M7 Rack Servers, also under a term-based license.You can order the Cisco NFVIS Smart License together with any Cisco UCS C-Series M5, M6, or M7 Rack Server through the Cisco Commerce Workspace (CCW) page. Once purchased, the license will appear as **Licenses Available to Use** in your Virtual Account on the Cisco Smart Software Manager (CSSM), Cisco's licensing platform.

Order Cisco NFVIS Smart License

You can order the Cisco NFVIS software Smart License along with the purchase of Cisco UCS C-Series M5, M6, and M7 Rack Servers. The Cisco NFVIS Smart License is a term-based license available for durations of 3, 5, or 7 years.

Note Cisco NFVIS Smart Licenses are sold in units corresponding to 32 physical CPU cores, excluding hyper-threaded cores.

Supported Devices For Cisco NFVIS Support For Cisco UCS Rack Servers

Supported Devices for Cisco NFVIS Smart Licensing on Cisco UCS C-Series Rack Servers

Table 44: Cisco UCS Rack Servers

Server Model	Minimum Supported Release	Supported PIDs
Cisco UCS C M6 Rack Server	Cisco NFVIS Release 4.13.1	• UCSC-C220-M6S • UCSC-C240-M6SX • UCSC-C240-M6S
Cisco UCS C M5 Rack Server	Cisco NFVIS Release 4.18.1	• UCSC-C220-M5SX • UCSC-C240-M5SX • UCSC-C240-M5S

Server Model	Minimum Supported Release	Supported PIDs
Cisco UCS C M7 Rack Server	Cisco NFVIS Release 4.18.1	• UCSC-C220-M7S • UCSC-C240-M7SX

Note For UCS C-Series M5 and M6 servers, if the rack servers have LOM (LAN on Motherboard) ports, the WAN bridge will be created on port GE0-0, which is a LOM port, and GE0-1 will be the LAN bridge on a LOM port. For UCS C-Series M7 Rack Servers, however, there are no LOM ports. Hence, the first slot in the chassis with a NIC inserted will have the WAN bridge and the LAN bridge.

Prerequisites For Cisco NFVIS Support For Cisco UCS C-Series Rack Servers

- You require an active Cisco Smart Account. For more information on Smart Account see, How to Request a Smart Account
- You need to purchase Cisco UCS C-Series Rack Servers along with Cisco NFVIS Smart License.

Restrictions For Cisco NFVIS Support For Cisco UCS C-Series Rack Servers

Cisco NFVIS is not engineered to produce any notifications pertaining to the expiration of the Cisco NFVIS license subscription. Inherently, it is not the Cisco NFVIS's role to monitor compliance. The sole responsibility for determining and tracking the expiration of the license subscription lies with the Cisco Smart Software Manager (CSSM).

Configure Cisco Smart Licensing for Cisco NFVIS

Smart Licensing for Cisco NFVIS operates in two modes:

1. Smart Mode
2. CSLU Mode

Smart Mode

In the smart mode operation, the Cisco UCS C-Series Rack Server has reachability to a Cisco Smart Server Manager (CSSM) server using Cisco NFVIS. The Cisco UCS C-Series Rack Server requires the following inputs to operate in the smart mode:

1. A unique token for customer virtual account on CSSM.

Note Generate a unique token using your Cisco Virtual Account on the Licensing Server (CSSM). For more information, see How can I create a token to register my device?

Provide the geneated token as the input to the Cisco UCS C-Series Rack Servers. The CSSM token is your password for the user account on CSSM. This input is mandatory for the Smart Mode to function and has no default values.

2. Transport Mode

   The Transport Mode specifies the type of connection existing between the device and CSSM (whether the device is directly connected/ Connected through CSLU). For the Smart Mode, the Transport Mode is called as Smart Transport. The default value for the transport mode is CSLU Mode.

3. Transport URL

   The Transport URL specifies the URL to reach a CSSM/CSLU server from the device. The default transport URL for Smart Mode is https://smartreceiver.cisco.com/licservice/license. The default transport URL for CSLU mode is available on online search engines.

4. Time interval between two licence usage announcements

   The Resource Usage Measurement (RUM) is sent periodically from the device to CSSM/CSLU which presents the License Usage pertaining to the NFVIS License Entitlement to the CSSM server. You can specify the periodic time interval as an input. The default value is 30 days.

Use the following methods to provide inputs to the Cisco UCS C-Series Rack Servers using Cisco NFVIS:

- Using the Cisco NFVIS CLI
- Using NETCONF/Rest APIs

When the device is provided with the above listed inputs for the smart mode of operation, the device initiates the Trust Handshake with the CSSM server. Once Trust Handshake is successful, the device starts reporting license usage according to the time interval specified.

Configure Smart Mode Using the CLI

```
configure terminal
license smart transport smart  smart-url
license smart transport smart  token
license smart transport smart  usage
```

Verify Smart Mode Using the CLI

Use the **show license opdata** command to verify the success of smart mode.

```
show license opdata
license opdata operational-status OK
license opdata usage-reporting last-ack-received None
license opdata usage-reporting usage-interval 1
license opdata usage-reporting next-ack-deadline "Oct 01 20:32:19 2023 UTC"
license opdata usage-reporting last-report-push None
license opdata usage-reporting next-report-push None
```

CSLU Mode

In the CSLU mode operation, Cisco UCS C-Series Rack Servers has reachability to a Cisco Smart Licensing Utility (CSLU) application.

- Ensure that you've logged into CSLU using your Cisco Smart Account on the CSSM.
- The license reporting in the CSLU mode happens at an interval of 30 days by default.
- The CSLU is the default Transport Mode and no token is required as an input.
- The default Transport URL for CSLU mode is http://cslu-local:8182/cslu/v1/pi
- If you require any change in the inputs other than the defaults, configure the inputs for CSLU mode.
- The CSLU Mode doesn't require any inputs and the CSLU Mode is the default mode of operation for the Cisco NFVIS on Cisco UCS C-Series Rack Servers.

Note

- There are no specific operations in terms of Cisco NFVIS Licensing with regards to the device reboot and upgrade. The Cisco NFVIS continues to operate with respect to the interactions established through the modes of operations.
- The licensing configuration and Smart Licensing Database aren't cleared from the device in case of a Factory Default Reset.

  License usage reporting to CSSM/CSLU resumes when the device regains network connectivity through a reconfig operation after a Factory Default Reset.
- To initiate an RMA for a device, manually release the licenses associated with the device from the CSSM, or use the Cisco NFVIS action command **`license smart release`**. The command frees up the license entitlements used by this device on CSSM. If the device continues to operate with Cisco NFVIS even after the license release, it will, after a set time interval, resend the usage report (RUM) to CSSM. This report indicates the correct number of license units consumed, which is reserved again on CSSM.
- While performing a device data backup, Cisco NFVIS backups the config to a file and sends the forced License Usage report to CSSM .

Monitor Cisco NFVIS Support for Cisco UCS C-Series Rack Server

Here is a list of show commands used to monitor Cisco NFVIS Smart Licensing for Cisco UCS C-Series Rack Servers along with some configuration examples:

- Use the **`show running-config license`** command in the Cisco NFVIS CLI to view the license information:

```
show running-config license
license smart
 transport smart
 usage interval 1
 token
```

"8dXijw3ROwmU/A2JWDctHBaLwAF6v+ESdMqZJqINCKQCYujbbJpMi8C3OWpjtW\nqS8FU2ljisfd49mV0qS+F92V8/uD3iUowGqjpWQ4FOcZb77OcjeAwHD
NMLNM0pR/+IF\nhNRgo4762u2ob6DjSr2YGcez8s7TMKQfjswUURDXYxYn1AG8VgSjs9RfPpFVi4Ly4BUFJg=="

- Use the **show license accounts** command in the Cisco NFVIS CLI to view the Cisco smart account information that the device is linked with:

```
show license accounts
license accounts VIRTUAL ACCOUNT:NFVIS-VA-1
                 SMART ACCOUNT:InternalTestDemoAccount8.cisco.com
```

- Use the **show license license-units-consumed** to view the units of entitlement used by the device on the CSSM:

```
show license license-units-consumed
license license-units-consumed 2
```

- Use the **show license opdata** command to view the licensing operational data:

```
show license opdata
license opdata operational-status OK
license opdata usage-reporting last-ack-received None
license opdata usage-reporting usage-interval 1
license opdata usage-reporting next-ack-deadline "Oct 01 20:32:19 2023 UTC"
license opdata usage-reporting last-report-push None
license opdata usage-reporting next-report-push None
```

- Use the **show license status** command to view the status of your license:

```
show license status
Current tenant: nfvis-single-tenant

                Smart Licensing is ENABLED

                Smart Licensing Policy:
                  Status: ENABLED

                Transport:
                  Type: SmartTransport
                  URL: https://smartreceiver-stage.cisco.com/licservice/license
                  Proxy:
                   Not configured

                Trust Code Installed: Jul 05 06:20:56 2023 UTC
                  Last Attempt Date: Jul 05 06:19:31 2023 UTC
                  Last Attempt Status: JobStatus.success
                  Last Attempt Result: success
                  Attempt In Progress: False
                  Next Attempt Date: None


                  Usage Reporting:
                  Last ACK received: None
                  Next ACK deadline: Oct 01 20:32:19 2023 UTC
                  Reporting push interval: 1
                  Next ACK push check: None
                  Next report push: None
                  Last report push: None
                  Last report file write: None


                Policy:
```

```
 Policy in use: None
 Policy name: Default Policy
 Reporting ACK required: yes
 Subscription:
  First report requirement (days): 90
  Ongoing reporting frequency (days): 90
  On change reporting (days): 90
 Perpetual:
  First report requirement (days): 365
  Ongoing reporting frequency (days): None
  On change reporting (days): 90
 Enforced:
  First report requirement (days): None
  Ongoing reporting frequency (days): None
  On change reporting (days): None
 Export:
  First report requirement (days): None
  Ongoing reporting frequency (days): None
  On change reporting (days): None
```

- Use the command **show license summary** to view the summary of your license

show license *summary*

```
license summary
                #### Current tenant: nfvis-single-tenant

                Smart Licensing is ENABLED

                Smart Licensing Policy:
                  Status: ENABLED

                Transport:
                  Type: SmartTransport
                  URL: https://smartreceiver-stage.cisco.com/licservice/license
                  Proxy:
                   Not configured

                Trust Code Installed: Jul 05 06:20:56 2023 UTC
                  Last Attempt Date: Jul 05 06:19:31 2023 UTC
                  Last Attempt Status: JobStatus.success
                  Last Attempt Result: success
                  Attempt In Progress: False
                  Next Attempt Date: None


                  Usage Reporting:
                  Last ACK received: None
                  Next ACK deadline: Oct 01 20:32:19 2023 UTC
                  Reporting push interval: 1
                  Next ACK push check: None
                  Next report push: None
                  Last report push: None
                  Last report file write: None
                Policy:
                  Policy in use: None
                  Policy name: Default Policy
                  Reporting ACK required: yes
                  Subscription:
                   First report requirement (days): 90
                   Ongoing reporting frequency (days): 90
                   On change reporting (days): 90
                  Perpetual:
                   First report requirement (days): 365
                   Ongoing reporting frequency (days): None
```

```
                    On change reporting (days): 90
                   Enforced:
                    First report requirement (days): None
                    Ongoing reporting frequency (days): None
                    On change reporting (days): None
                   Export:
                    First report requirement (days): None
                    Ongoing reporting frequency (days): None
                    On change reporting (days): None


                 License Usage
                 =============
                   Name: None
                   Short-name-tag: NFVIS_CORE_UCSC
                   Name-tag:
regid.2023-04.com.cisco.NFVIS_CORE_UCSC,1.0_81ac6671-e3e7-489e-ae14-9610ae3ccdaf
                   Description: None
                   Count: 2
                   Version: None
                   Status: IN USE
                   Enforcement_type: NOT ENFORCED
                   Feature: None
                   Feature description: None
```

- Use the **show license usage** command to view your license usage:

```
show license usage
license usage
              License Usage
              =============
                Name: None
                Short-name-tag: NFVIS_CORE_UCSC
                Name-tag:
regid.2023-04.com.cisco.NFVIS_CORE_UCSC,1.0_81ac6671-e3e7-489e-ae14-9610ae3ccdaf
                Description: None
                Count: 2
                Version: None
                Status: IN USE
                Enforcement_type: NOT ENFORCED
                Feature: None
                Feature description: None
```

- The license tech support is a part of Cisco NFVIS tech support.

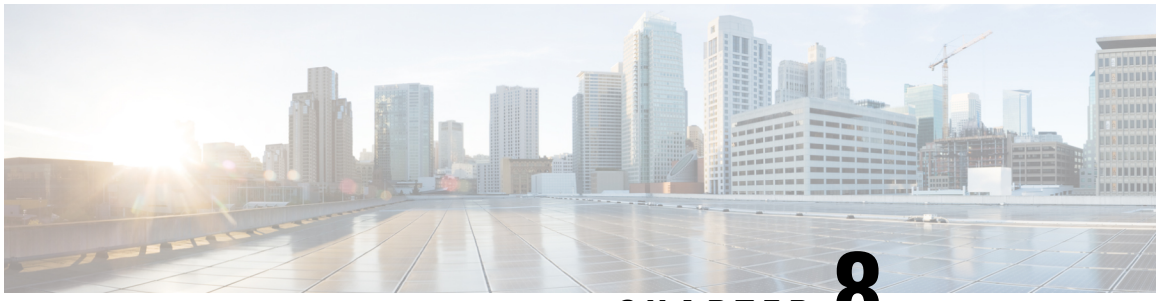
Action Commands

Use the following actions commands that can help you release, sync and trust Cisco NFVIS licences:

- **license smart release**: Use this command to send a "License Usage 0" message to CSSM. This action prompts the CSSM to release the license entitlement associated with your Cisco UCS C Rack servers. If the device continues to operate with Cisco NFVIS after the license release, a usage report, also known as RUM, is sent to CSSM after a predetermined time interval. This report will indicate the actual number of license units consumed, which will then be reserved again on CSSM. Following a license release, a notification and system logging warning is issued, urging you to cease using Cisco NFVIS as the license has been released. You receive this notification every 8 hours during the 24 hours after a license release. If you continue to use Cisco NFVIS beyond this 24-hour period, a RUM report

will be generated reflecting the appropriate number of license units consumed by Cisco NFVIS. This report is sent to the Licensing server based on the set periodic interval, after which the licensing server will reserve the correct number of license units for the device again.

- **`license smart sync`**: Report a license usage to CSSM using this action command.
- **`license smart trust`**: Initiate the establishment of trust between the device and the CSSM. This action can prove beneficial in various scenarios, including but not limited to the following:
  1. If you delete the product instance from CSSM but wish to continue with Cisco NFVIS Licensing.
  2. If you transfer your licenses from one Virtual Account to another.
  3. If there is an asynchrony between the licensing state on CSSM and on the device.
  4. If the licensing certificates on the device reach their expiry date.



CHAPTER 8

Secure Overlay and Single IP Configuration

- Secure Overlay, on page 197
- Single Public IP Address and Secure Overlay, on page 207
- Single IP Address Without Secure Overlay, on page 212

Secure Overlay

Note The Secure Overlay feature will be deprecated and its last supported release is Cisco NFVIS Release 4.18.1 For future releases, this feature will no longer be supported.

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.

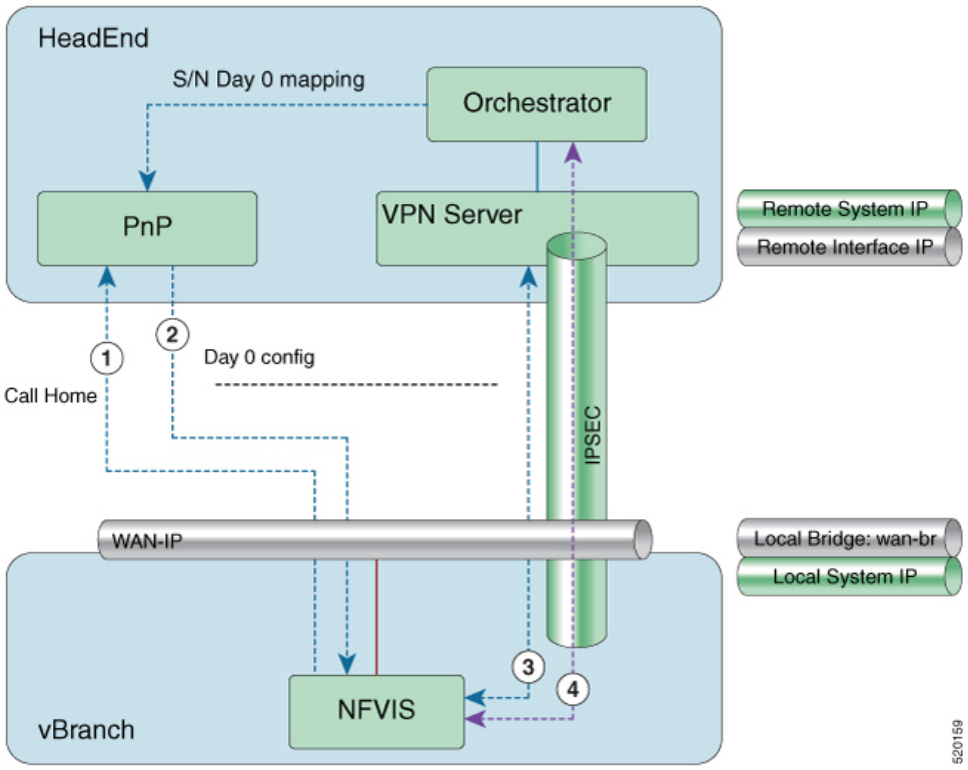
- Remote system IP address:
    - IP address
    - Introduced in NFVIS 4.4.1 release - Active-Standby headend VPN responder scenarios is supported. You can configure a list of Remote system IP addresses, each of which is corresponding to a tunnel IP address on a remote VPN responder.
    - Introduced 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 192.0.2.1
    local-system-ip-bridge int-mgmt-net
    local-id mail@gmail.com
    remote-interface-ip-addr nfvisoverlay.example.com
    remote-system-ip-addr [ 192.0.2.111 192.0.2.112 ]
    psk local-psk Cisco1234Admin
    psk remote-psk Cisco1234Admin
     !

confirgure terminal
secure-overlay myconn
local-system-ip-addr 192.0.2.2
local-system-ip-subnet 192.0.2.0/24
local-system-ip-bridge int-mgmt-net
local-id AxxxY@example.com
remote-interface-ip-addr C*****d.example.com
remote-system-ip-addr [ 192.0.2.113 166.34.121.112 ]
remote-system-ip-subnet [ 192.0.2.112/32 192.0.2.112/32 ]
remote-id [
CN=vbranch,unstructuredAddress=10.30.1.114,unstructuredName=csr-vpn-srvr-02.example.com
CN=vbranch,unstructuredAddress=10.30.1.153,unstructuredName=csr-vpn-srvr-03.example.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    192.0.2.2  10.30.1.153  192.0.2.112  192.0.2.112/32
  "CN=vbranch,unstructuredAddress=10.30.1.153,unstructuredName=csr-vpn-srvr-03.example.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>192.0.2.2</local-system-ip-addr>
      <local-system-ip-subnet>192.0.2.0/24</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>192.0.2.112</remote-system-ip-addr>
      <remote-system-ip-addr>192.0.2.112/32</remote-system-ip-addr>
      <remote-system-ip-subnet>192.0.2.112/32</remote-system-ip-subnet>
      <remote-system-ip-subnet>192.0.2.112/32</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 192.0.2.103 192.0.2.90
 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 10.90.90.1 255.255.255.255
interface Loopback2
 ip vrf forwarding private-vrf
 ip address 10.91.91.1 255.255.255.0
interface Loopback3
 ip vrf forwarding private-vrf
 ip address 10.90.90.201 255.255.255.128
interface Loopback1001
 description MNSO Interface and Tunnel Loopback
 ip vrf forwarding private-vrf
 ip address 192.0.2.114 255.255.255.255
interface GigabitEthernet1
 ip vrf forwarding private-vrf
 ip address 10.92.92.1 255.255.255.0
 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 10.91.91.1 255.255.255.0
 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 10.12.12.1.

```
<secure-overlay>
   <name>mgmthub</name>
   <local-bridge>wan-br</local-bridge>
   <local-system-ip-addr>10.12.12.1</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>10.12.12.1</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>10.12.12.1</local-system-ip-addr>
   <local-system-ip-bridge>int-mgmt-net</local-system-ip-bridge>
   <local-id>branch101@example.com</local-id>
   <remote-interface-ip-addr> 192.0.2.75</remote-interface-ip-addr>
   <remote-system-ip-addr> 192.168.1.90</remote-system-ip-addr>
   <remote-id>CN=vbranch, unstructuredAddress=192.0.2.75,
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://192.0.2.75:80
 ip-address 192.0.2.75
 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 192.0.2.75
 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 192.0.2.75 255.255.255.0
 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
```

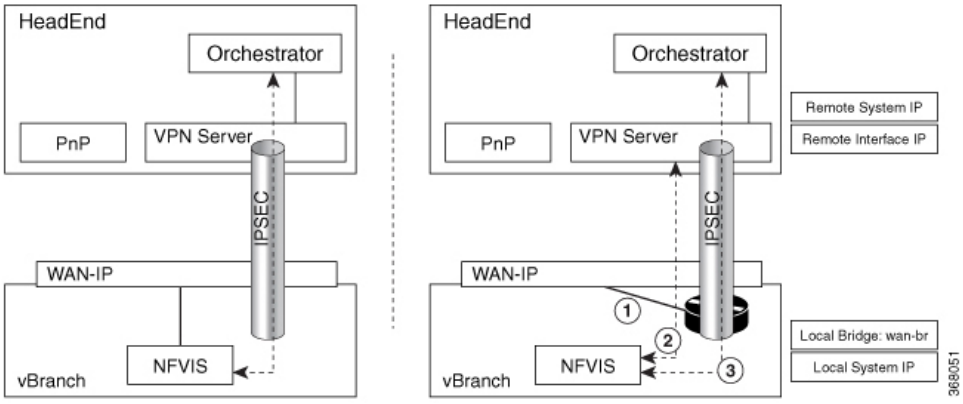
Use the **show vm_lifecycle deployments all | include Name** command to find out the Deployment Name and VM Group Name.

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 2: Example of Setting IPSec Tunnel in Single IP mode



After secure overlay over WAN is established, the orchestrator sends requests to configure single IP mode and deploy the guest router that takes the public IP address.

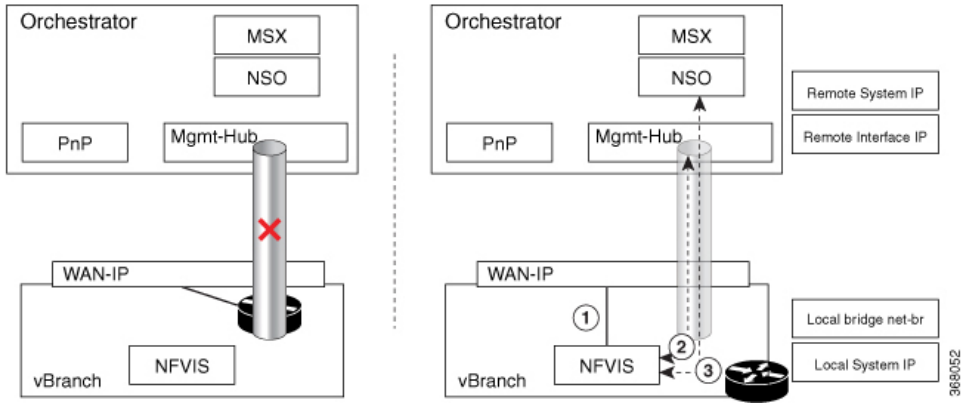
1. NFVIS deploys the VM with specified bootstrap and Day-0 configuration. NFVIS takes down the current IPSec tunnel and releases the public IP address.
2. The VM takes the public IP address when it is in active state. NFVIS sets up the IPSec tunnel again with the remote management hub.

3. After the IPSec tunnel is up, the orchestrator can connect to NFVIS through its system IP address and manage NFVIS over the IPSec tunnel.

In single IP mode, NFVIS monitors the guest VM taking the public IP address. NFVIS takes WAN IP address back when the guest VM is:

- In error state.
- Stopped through vmAction.
- Undeployed.

Figure 3: Example of NFVIS Handling Failure



1. NFVIS takes WAN IP address.
2. NFVIS sets up IPSec tunnel to the management hub.
3. When IPSec tunnel is up, the VPN server can connect to NFVIS through its system IP address and manage NFVIS over the IPSec tunnel.

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 10.12.12.0
netmask 255.255.255.0
gateway 10.12.12.1
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 192.0.2.17 255.255.255.0
ip nat outside
negotiation auto
!
ip nat inside source list NAT interface GigabitEthernet2 overload
ip route 0.0.0.0.0.0.0.0 192.0.2.1
ip route 10.19.18.0 255.255.255.0 10.12.12.1
!
ip access-list standard NAT
permit 10.12.12.0.0.0.0.25
```

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 10.12.12.1
!
ip access-list standard NAT
permit 10.12.12.0.0.0.0.25
```

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.

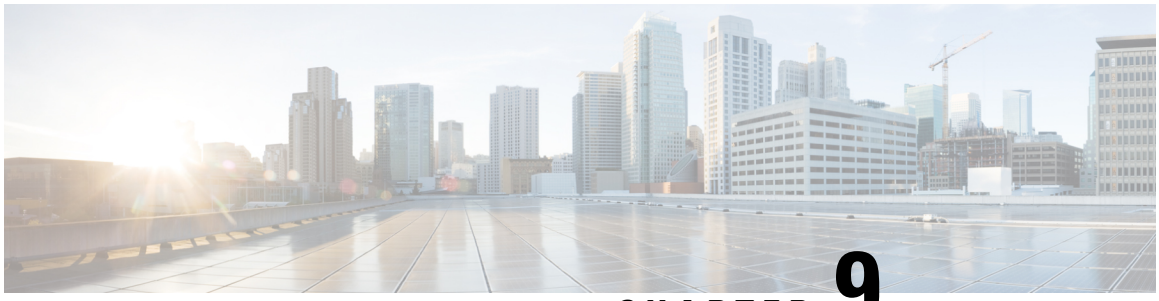
The following example is a sample bootstrapping configuration:

192.0.2.7/24 is the single public IP address that is originally used by NFVIS and later by the VM. 192.0.2.1 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 192.0.2.17 255.255.255.0
ip nat outside
negotiation auto
!
ip nat inside source list NAT interface GigabitEthernet2 overload
ip route 0.0.0.0 0.0.0.0 192.0.2.1
!
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 9

BGP Support on NFVIS

Table 45: Feature History

Feature Name	Release Information	Description
BGP Support on Remote Subnets Over IPSec.	NFVIS 4.4.1	This feature allows the NFVIS system to learn routes that are announced from the remote BGP neighbor and apply the learnt routes to the NFVIS system.
BGP Support Announcing Local Subnets (Route Distribution)	NFVIS 3.10.1	This feature allows you to announce or withdraw NFVIS local routes to the remote BGP neighbor using route distribution.

Border Gateway Protocol (BGP) is the dynamic routing protocol to exchange route information between BGP autonomous systems.

The NFVIS BGP feature works together with remote BGP router. This feature allows NFVIS system to learn routes announced from the remote BGP neighbor and apply the learnt routes to the NFVIS system. This feature also allows you to announce or withdraw NFVIS local routes from the remote BGP neighbor.

Starting from NFVIS 4.4.1 release, NFVIS BGP feature works with the secure overlay feature to learn routes from the BGP neighbor over a secure overlay tunnel. These learnt routes or subnets are added into the NFVIS routing table for the secure tunnel, which makes the routes accessible over the tunnel.

- Configure BGP on NFVIS, on page 215
- Route Distribution, on page 218
- BGP Route Annoucement over MPLS or IPSec, on page 219

Configure BGP on NFVIS

A BGP neighbor can be configured using a neighbor IP address or a name string.

If a BGP neighbor is specified using a name string, it must be used in conjunction with the secure overlay bgp-neibhor-name field. A BGP session is established over the secure overlay tunnel. If the neighbor name matches the bgp-neighbor-name field configured in secure-overlay configuration, then NFVIS will determine the active remote system IP address used for IPSec connection and replace the neighbor name with that IP.

This will establish a BGP neighbor session with that IP address. For more information on how to configure secure overlay with BGP name, see Secure Overlay and Single IP Configuration, on page 197.

If a BGP neighbor is specified using an IP address which is headend VPN responder's tunnel IP address, which is the same as the IP address of a a headend VPN responder tunnel, a BGP session is established over the secure overlay tunnel.

This example shows how to create or update BGP configuration for a neighbor with a specified name string:

```
config terminal
router bgp 200
    neighbor csrbgp remote-as 65000
        commit
```

This example shows how to create or update BGP configuration with a specified neighbor IP address:

```
config terminal
router bgp 200
     neighbor 166.34.121.112 remote-as 65000
         exit
     neighbor 166.35.121.112 remote-as 65000
         commit
```

This example shows how to delete BGP configurations:

```
no router bgp 200
commit
```

The following table provides the syntax description for each parameter in the commands mentioned in the examples above:

Property	Type	Description	Mandatory
as	Uint32	Local BGP AS number	Yes
router-id	IPv4	H.H.H.H: IPv4 address for local system	No
neighbor	list	Neighbor list	Yes
remote-ip	String	IPv4 address or Secure Overlay BGP neighbor name for BGP neighbor system	Yes
remote-as	Uint32	Remote BGP AS number	Yes
description	String	Description of neighbor	No

The following example displays the BGP session details:

```
nfvis# support show bgp

BIRD 1.6.8 ready.
name    proto    table    state    since    info
bgp1    BGP      bgptable  UP     23:53:18  Established
 Preference:     100
 Input filter:   ACCEPT
```

```
Output filter: Accept
Import limit:  15
   Action:     restart
Routes:        1 import, 0 exported, 1 preferred
Route change stats:     recieved  rejected  filtered  ignored  accepted
  Import updates:              1         0         0        0         1
  Import withdraws:            0         0       ---        0         0
  Export updates:              1         1         0      ---         0
  Export withdraws:            0       ---       ---      ---         0
BGP state:     Established
  Neigbour addres: 166.34.121.112
  Neigbour AS:     65000
  Neigbour ID:     166.34.121.112
  Neigbour caps:   refresh enhanced-refresh AS4
  Session:         external multihop AS4
  Source Address:  112.112.112.1
  Route limit:     1/15
  Hold timer:      204/240
  Keepalive timer: 65/80
```

The following example displays the BGP routes learnt through BGP:

```
nfvis# support show bgp route

BIRD 1.6.8 ready.
91.91.91.0/24      dev ipsec0 [bgp1 23:53:18 from 166.34.121.112] (100) [AS65000?]
```



Note NFVIS can learn up to 15 prefixes.

BGP Neighbor Configuration Example

```
router bgp 65000
 bgp router-id 166.34.121.112
 bgp always-compare-med
 bgp log-neighbor-changes
 bgp deterministic-med
 bgp listen range 112.112.0.0/16 peer-group uCPEs
 bgp listen range 90.90.90.0/24 peer-group uCPEs
 bgp listen range 10.20.0.0/24 peer-group uCPEs
 bgp listen limit 255
 no bgp default ipv4-unicast
 !
 address-family ipv4 vrf private-vrf
  redistribute connected
  redistribute static
  neighbor uCPEs peer-group
  neighbor uCPEs remote-as 200
  neighbor uCPEs ebgp-multihop 10
  neighbor uCPEs timers 610 1835
  neighbor uCPEs prefix-list allow-list out
 exit-address-family
```

Route Distribution

Note The Route Distribution feature will be deprecated and its last supported release is Cisco NFVIS Release 4.18.1. For future releases, this feature will no longer be supported.

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

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

To configure or update route distribution:

```
configure terminal
router bgp 172.25.221.17local-bridge wan-br local-as 45.45remote-as 65000 network-subnet
12.12.12.0/24
commit
```

Table 46: Property Description

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

Property	Type	Description	Mandatory
remote-as		Remote autonomous system number. It can be in following two formats: <decimal number, 1.0 .. 65535.65535><unsignedInt, 1 .. 4294967295>	Yes
router-id	IPv4	Local router ID	No
network-subnet		List of network subnet to be announced.	Yes
subnet	IPv4 prefix	Network subnet to be announced H.H.H.H/N	Yes
next-hop	IPv4	IPv4 address of next hop. Default local-address or IP address of local-bridge.	No

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

Remote BGP Router Configuration Example

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

This example shows the configuration on a remote BGP router.

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

BGP Route Annoucement over MPLS or IPSec

Table 47: Feature History

Feature Name	Release Information	Description
BGP Route Annoucement over MPLS or IPSec	NFVIS 4.5.1	This feature allows you to configure NFVIS to announce routes through BGP over MPLS. NFVIS allows the routes learnt through BGP available over IPSec tunnel over MPLS connection.

With this feature enhancement, the existing routes learnt through BGP over IPSec tunnel are now allowed over MPLS connection. Additionally, NFVIS can now announce routes through BGP, using the same **router**

bgp command that is used for learning routes over BGP. For more information on this command, see the Cisco IOS XE router bgp command.

You can pair the secure overlay configurations to announce NFVIS routes over BGP through IPSec tunnel. The existing router bgp configurations can be updated to add the route announcement feature. Make sure that you remove the existing route distribution configurations before you configure the **router bgp** command.

The following example shows how to configure the announcement of 10.20.0.0/24 subnet over BGP.

```
router bgp 65000
  neighbor 172.25.221.17 remote-as 65001
  address-family ipv4 unicast
    network 10.20.0.0 mask 255.255.255.0
    neighbor 172.25.221.17 activate
```

The following example shows how to remove the announcement of 10.20.0.0/24 subnet from BGP.

```
router bgp 65000
  address-family ipv4 unicast
    no network 10.20.0.0 mask 255.255.255.0
```

The following example shows how to remove a neighbor from the IPv4 address family, and disable route announcements for the same neighbor.

```
router bgp 65000
  address-family ipv4 unicast
    no neighbor 172.25.221.17 activate
```

To view the local BGP status for BGP over MPLS use the **show bgp ipv4 unicast** command.

```
nfvis# show bgp ipv4 unicast

Family Transmission Router ID       Local AS Number
ipv4   unicast      10.20.0.1       65000
```

To view the BGP neighbor status for BGP over MPLS use the **show bgp ipv4 unicast summary** command.

```
nfvis# show bgp ipv4 unicast summary

Neighbor        IP Version AS Number Up/Down
172.25.221.17   4          65001     up
```

To view the BGP learned or announced routes for BGP over MPLS use the **show bgp ipv4 unicast route** command.

```
nfvis# show bgp ipv4 unicast route

Network            Next-Hop        Metric LocPrf Path
10.30.30.0/24      172.25.221.17   0      100    65001 ?
10.40.40.0/24      172.25.221.17   0      100    65001 ?
10.20.0.0/24       0.0.0.0
```

To view the local BGP status for BGP over IPSec tunnel use the **show bgp vpnv4 unicast** command.

```
nfvis# show bgp vpnv4 unicast

Family Transmission Router ID       Local AS Number
vpnv4  unicast      10.20.0.1       200
```

To show BGP neighbor status for BGP over IPSec tunnel:

```
nfvis# show bgp vpnv4 unicast summary

Neighbor        IP Version AS Number Up/Down
10.90.90.1      4          65000     up
```

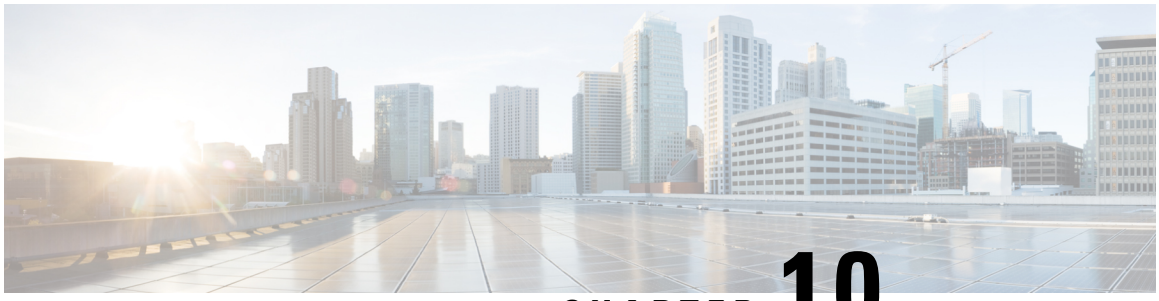
To show BGP learned/announced routes for BGP over IPSec tunnel:

```
nfvis# show bgp vpnv4 unicast route

Network            Next-Hop        Metric LocPrf Path
10.91.91.0/24      10.90.90.1      0      100    65000 ?
10.92.92.0/24      10.90.90.1      0      100    65000 ?
10.20.0.0/24       0.0.0.0                       i
```

Note When you configure BGP route announcement over IPSec tunnel, ensure that you configure secure overlay to use the virtual IP address for the local tunnel IP address (**no local-system-ip-addr configured**).

Note When you configure BGP route announcement, the only configurable address-family or transmission combination is **ipv4 unicast** for both IPSec and MPLS. To view the BGP status, the configurable address-family or transmission for IPSec is **vpnv4 unicast** and for MPLS is **ipv4 unicast**.

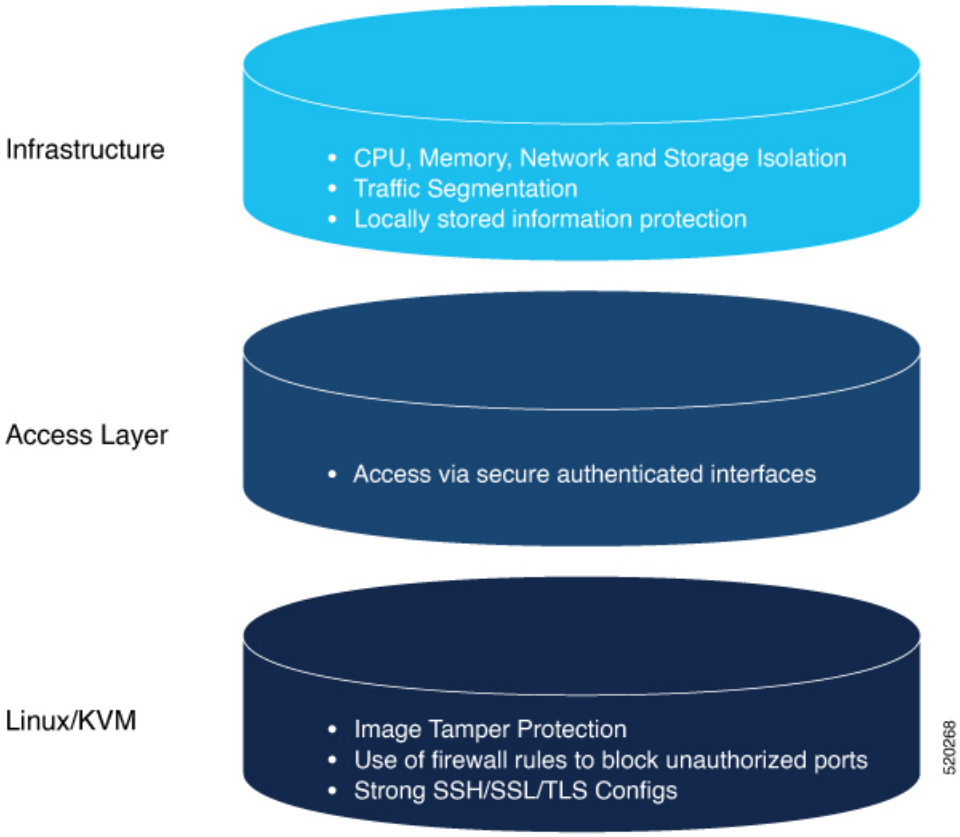


CHAPTER 10

Security Considerations

This chapter describes the security features and considerations in NFVIS. It gives a high-level overview of security related components in NFVIS to plan a security strategy for deployments specific to you. It also has recommendations on security best practices for enforcing the core elements of network security.

The NFVIS software has security embedded right from installation through all software layers. The subsequent chapters focus on these out-of-the-box security aspects such as credential management, integrity and tamper protection, session management, secure device access and more.



- Installation, on page 224
- Secure Unique Device Identification, on page 225
- Device Access, on page 226

- Infrastructure Management Network, on page 244
- Locally Stored Information Protection, on page 245
- File Transfer, on page 246
- Logging, on page 246
- Virtual Machine security, on page 247
- VM Isolation and Resource provisioning, on page 248
- Secure Development Lifecycle, on page 251

Installation

To ensure that the NFVIS software has not been tampered with , the software image is verified before installation using the following mechanisms:

Image Tamper Protection

NFVIS supports RPM signing and signature verification for all RPM packages in the ISO and upgrade images.

RPM Signing

All RPM packages in the Cisco Enterprise NFVIS ISO and upgrade images are signed to ensure cryptographic integrity and authenticity. This guarantees that the RPM packages have not been tampered with and the RPM packages are from NFVIS. The private key used for signing the RPM packages is created and securely maintained by Cisco.

RPM Signature Verification

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

Scenario	Description
Cisco Enterprise NFVIS 3.7.1 and later installations	If the signature verification fails while installing Cisco Enterprise NFVIS, the installation is aborted.
Cisco Enterprise NFVIS upgrade from 3.6.x to Release 3.7.1	The RPM signatures are verified when the upgrade is being performed. If the signature verification fails, an error is logged but the upgrade is completed.
Cisco Enterprise NFVIS upgrade from Release 3.7.1 to later releases	The RPM signatures are verified when the upgrade image is registered. If the signature verification fails, the upgrade is aborted.

Image Integrity Verification

RPM signing and signature verification can be done only for the RPM packages available in the Cisco NFVIS ISO and upgrade images. To ensure the integrity of all the additional non-RPM files available in the Cisco NFVIS ISO image, a hash of the Cisco NFVIS ISO image is published along with the image. Similarly, a hash of the Cisco NFVIS upgrade image is published along with the image. To verify that the hash of Cisco

NFVIS ISO image or upgrade image matches the hash published by Cisco, run the following command and compare the hash with the published hash:

```
% /usr/bin/sha512sum <ImageFile>
c2122783efc18b039246ae1bcd4eec4e5e027526967b5b809da5632d462dfa6724a9b20ec318c74548c6bd7e9b8217ce96b5ece93dcddf74fda5e01bb382ad607
  <ImageFile>
```

ENCS Secure Boot

Secure boot is part of the Unified Extensible Firmware Interface (**UEFI**) standard which ensures that a device boots only using a software that is trusted by the Original Equipment Manufacturer (OEM). When NFVIS starts, the firmware checks the signature of the boot software and the operating system. If the signatures are valid, the device boots, and the firmware gives the control to the operating system.

Secure boot is available on the ENCS but is disabled by default. Cisco recommends you to enable secure boot. For more information, see Secure Boot of Host.

Secure Unique Device Identification

NFVIS uses a mechanism known as Secure Unique Device Identification (SUDI), which provides it with an immutable identity. This identity is used to verify that the device is a genuine Cisco product, and to ensure that the device is well-known to the customer's inventory system.

The SUDI is an X.509v3 certificate and an associated key-pair which are protected in hardware. The SUDI certificate contains the product identifier and serial number and is rooted in Cisco Public Key Infrastructure. The key pair and the SUDI certificate are inserted into the hardware module during manufacturing, and the private key can never be exported.

The SUDI-based identity can be used to perform authenticated and automated configuration using Zero Touch Provisioning (ZTP). This enables secure, remote on-boarding of devices, and ensures that the orchestration server is talking to a genuine NFVIS device. A backend system can issue a challenge to the NFVIS device to validate its identity and the device will respond to the challenge using its SUDI based identity. This allows the backend system to not only verify against its inventory that the right device is in the right location but also provide encrypted configuration that can only be opened by the specific device, thereby ensuring confidentiality in transit.

The following workflow diagrams illustrate how NFVIS uses SUDI:

Figure 4: Plug and Play (PnP) Server authentication

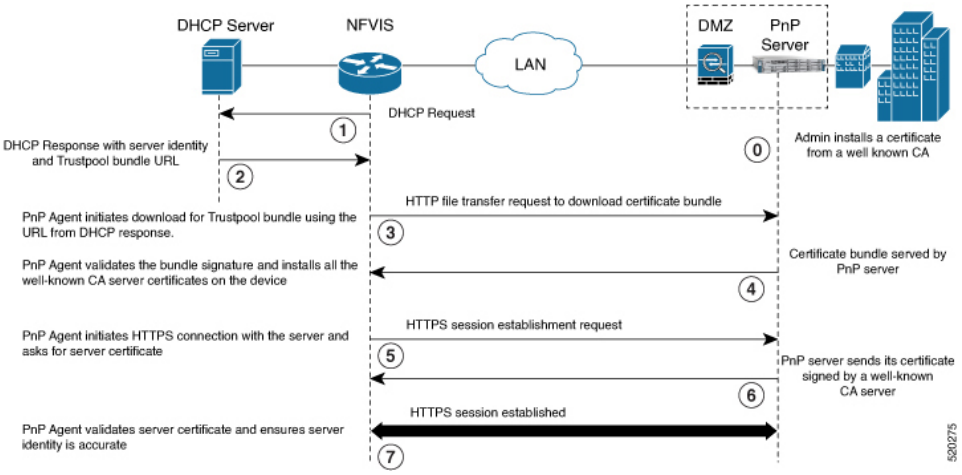
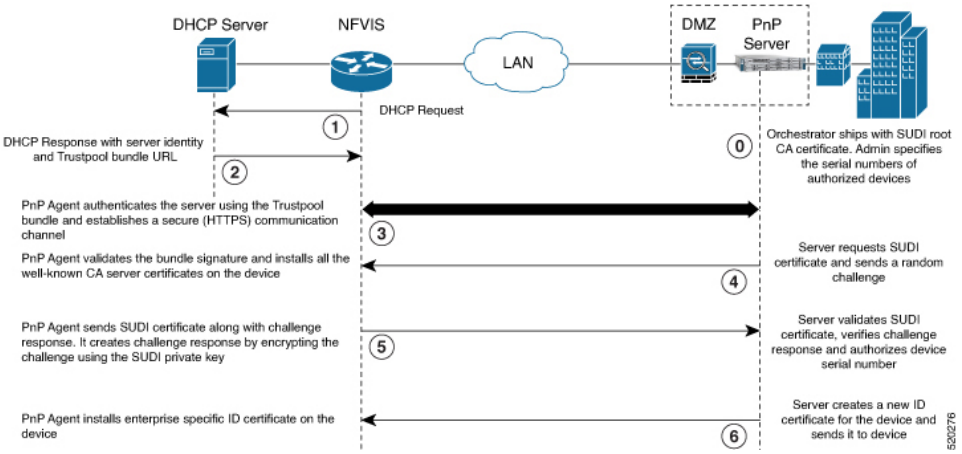


Figure 5: Plug and Play Device Authentication and Authorization



Device Access

NFVIS provides different access mechanisms including console as well as remote access based on protocols such as HTTPS and SSH. Each access mechanism should be carefully reviewed and configured. Ensure that only the required access mechanisms are enabled and that they are properly secured. The key steps to securing both interactive and management access to NFVIS are to restrict the device accessibility, restrict the capabilities of the permitted users to what is required, and restrict the permitted methods of access. NFVIS ensures that the access is only granted to authenticated users and they can perform just the authorized actions. Device access is logged for auditing and NFVIS ensures the confidentiality of locally stored sensitive data.

It is critical to establish the appropriate controls in order to prevent unauthorized access to NFVIS. The following sections describe the best practices and configurations to achieve this:

Enforced Password Change at First Login

Default credentials are a frequent source of product security incidents. Customers often forget to change the default login credentials leaving their systems open to attack. To prevent this, the NFVIS user is forced to change the password after the first login using the default credentials (username: admin and password Admin123#).

For more information, see Accessing NFVIS, on page 15.

Restricting Login Vulnerabilities

You can prevent the vulnerability to dictionary and Denial of Service (DoS) attacks by using the following features.

Enforcement of Strong password

An authentication mechanism is only as strong as its credentials. For this reason, it is important to ensure users have strong passwords. NFVIS checks that a strong password is configured as per the following rules:

Password must contain:

- At least one uppercase character
- At least one lowercase character
- At least one number
- At least one of these special characters: hash (#), underscore (_), hyphen (-), asterisk (*), or question mark (?)
- Seven characters or more
- The password length should be between 7 and 128 characters.

Configuring Minimum Length for Passwords

Lack of password complexity, particularly password length, significantly reduces the search space when attackers try to guess user passwords, making brute-force attacks much easier.

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

CLI:

```
nfvis(config)# rbac authentication min-pwd-length 9
```

API:

```
/api/config/rbac/authentication/min-pwd-length
```

Configuring Password Lifetime

The password lifetime determines how long a password can be used before the user is required to change it.

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

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



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

CLI:

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

API:

```
/api/config/rbac/authentication/password-lifetime/
```

Limit previous password reuse

Without preventing the use of previous passphrases, password expiry is largely useless since users can simply change the passphrase and then change it back to the original.

NFVIS checks that the new password is not the same as one of the 5 previously used passwords. One exception to this rule is that the admin user can change the password to the default password even if it was one of the 5 previously used passwords.

Restrict Frequency of login attempts

If a remote peer is allowed to login an unlimited number of times, it may eventually be able to guess the login credentials by brute force. Since passphrases are often easy to guess, this is a common attack. By limiting the rate at which the peer can attempt logins, we prevent this attack. We also avoid spending the system resources on unnecessarily authenticating these brute-force login attempts which could create a Denial of Service attack.

NFVIS enforces a 5 minute user lockdown after 10 failed login attempts.

Disable inactive user accounts

Monitoring user activity and disabling unused or stale user accounts helps to secure the system from insider attacks. The unused accounts should eventually be removed.

The admin user can enforce a rule to mark unused user accounts as inactive and configure the number of days after which an unused user account is marked as inactive. Once marked as inactive, that user cannot login to the system. To allow the user to login to the system, the admin user can activate the user account.



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

The following CLI and API can be used to configure the enforcement of account inactivity.

CLI:

```
configure terminal
rbac authentication account-inactivity enforce true inactivity-days 30
commit
```

API:

```
/api/config/rbac/authentication/account-inactivity/
```

The default value for inactivity-days is 35.

Activating an Inactive User Account

The admin user can activate the account of an inactive user using the following CLI and API:

CLI:

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

API:

```
/api/operations/rbac/authentication/users/user/username/activate
```

Enforce Setting of BIOS and CIMC Passwords

Table 48: Feature History Table

Feature Name	Release Information	Description
Enforce Setting of BIOS and CIMC Passwords	NFVIS 4.7.1	This feature enforces the user to change the default password for CIMC and BIOS.

Restrictions for Enforcing Setting of BIOS and CIMC Passwords

- This feature is only supported on Cisco Catalyst 8200 UCPE and Cisco ENCS 5400 platforms.
- This feature is only supported on a fresh install of NFVIS 4.7.1 and later releases. If you upgrade from NFVIS 4.6.1 to NFVIS 4.7.1, this feature is not supported and you are not prompted to reset the BIOS and CIMS passwords, even if the BIOS and CIMC passwords are not configured.

Information About Enforcing Setting of BIOS and CIMC Passwords

This feature addresses a security gap by enforcing the resetting of the BIOS and CIMC passwords after a fresh install of NFVIS 4.7.1. The default CIMC password is **password** and the default BIOS password is no password.

In order to fix the security gap, you are enforced to configure the BIOS and CIMC passwords in ENCS 5400. During a fresh install of NFVIS 4.7.1, if the BIOS and CIMC passwords have not been changed and still have

the default passwords, then you are prompted to change both the BIOS and CIMC passwords. If only one of them requires reset, you are prompted to reset the password for only that component. Cisco Catalyst 8200 UCPE requires only the BIOS password and hence only the BIOS password reset is prompted, if it has not already been set.



Note If you upgrade from any previous release to NFVIS 4.7.1 or later releases, you can change the BIOS and CIMC passwords using the **hostaction change-bios-password newpassword** or **hostaction change-cimc-password newpassword** commands.

For more information about BIOS and CIMC passwords, see BIOS and CIMC Password.

Configuration Examples for Enforced Resetting of BIOS and CIMC Passwords

1. When you install NFVIS 4.7.1, you must first reset the default admin password.

```
Cisco Network Function Virtualization Infrastructure Software (NFVIS)

NFVIS Version: 99.99.0-1009

Copyright (c) 2015-2021 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 10.24.109.102 using ssh on nfvis
admin logged with default credentials
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
```

2. On Cisco Catalyst 8200 UCPE and Cisco ENCS 5400 platforms when you do a fresh install of NFVIS 4.7.1 or later releases, you must change the default BIOS and CIMC passwords. If the BIOS and CIMC passwords are not previously configured, the system prompts you the reset the BIOS and CIMC passwords for Cisco ENCS 5400 and only the BIOS password for Cisco Catalyst 8200 UCPE.

```
New admin password is set

Please provide the BIOS 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 #, @ or _
            5. Length should be between 8 and 20 characters
            6. Should not contain any of the following strings (case sensitive): bios
            7. First character cannot be a #
```

```
Please reset the BIOS password :
Please reenter the BIOS password :
Please provide the CIMC 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 #, @ or _
            5. Length should be between 8 and 20 characters
            6. Should not contain any of the following strings (case sensitive): admin
Please reset the CIMC password :
Please reenter the CIMC password :
```

Verify BIOS and CIMC Passwords

To verify if the BIOS and CIMC passwords are changed succesfully, use the **show log nfvis_config.log | include BIOS** or **show log nfvis_config.log | include CIMC** commands:

```
nfvis# show log nfvis_config.log | include BIOS
2021-11-16 15:24:40,102 INFO     [hostaction:/system/settings]  []  BIOS password change
is successful
```

You can also download the nfvis_config.log file and verify if the passwords are reset successfully.

Integration with external AAA servers

Users login to NFVIS through ssh or the Web UI. In either case, users need to be authenticated. That is, a user needs to present password credentials in order to gain access.

Once a user is authenticated, all operations performed by that user need to be authorized. That is, certain users may be allowed to perform certain tasks, whereas others are not. This is called authorization.

It is recommended that a centralized AAA server be deployed to enforce per-user, AAA-based login authentication for NFVIS access. NFVIS supports RADIUS and TACACS protocols to mediate network access. On the AAA server, only minimum access privileges should be granted to authenticated users according to their specific access requirements. This reduces the exposure to both malicious and unintentional security incidents.

For more information on external authentication, see Configuring RADIUS, on page 38 and Configuring a TACACS+ Server, on page 40.

Authentication Cache for External Authentication Server

Feature Name	Release Information	Description
Authentication Cache for External Authentication Server	NFVIS 4.5.1	This feature supports TACACS authentication through OTP on NFVIS portal.

The NFVIS portal uses the same One-Time Password (OTP) for all API calls after the initial authentication. The API calls fail as soon as the OTP expires. This feature supports TACACS OTP authentication with the NFVIS portal.

After you have successfully authenticated through the TACACS server using an OTP, NFVIS creates a hash entry using the username and the OTP and stores this hash value locally. This locally stored hash value has

an expiration time stamp associated with it. The time stamp has the same value as the SSH session idle timeout value which is 15 minutes. All the subsequent authencation requests with the same username are authenticated against this local hash value first. If the authentication fails with the local hash, NFVIS authenticates this request with TACACS server and creates a new hash entry when the authentication is successful. If a hash entry already exists, its time stamp is reset to 15 minutes.

If you are removed from the TACACS server after successfully logging into the portal, you can continue to use the portal until the hash entry in NFVIS expires.

When you explicitly log out from the NFVIS portal or are logged out due to idle time, the portal calls a new API to notify NFVIS backend to flush the hash entry. The authentication cache and all of its entries are cleared out after NFVIS reboot, factory reset, or upgrade.

Role Based Access Control

Limiting network access is important to organizations that have many employees, employ contractors or permit access to third parties, such as customers and vendors. In such a scenario, it is difficult to monitor network access effectively. Instead, it is better to control what is accessible, in order to secure the sensitive data and critical applications.

Role-based access control (RBAC) is a method of restricting network access based on the roles of individual users within an enterprise. RBAC lets users access just the information they need, and prevents them from accessing information that doesn't pertain to them.

An employee's role in the enterprise should be used to determine the permissions granted, in order to ensure that employees with lower privileges can't access sensitive information or perform critical tasks.

The following user roles and privileges are defined in NFVIS

User Role	Privilege
Administrators	Can configure all available features and perform all tasks including changing of user roles. The administrator cannot delete basic infrastructure that is fundamental to NFVIS. The Admin user’s role cannot be changed; it is always "administrators".
Operators	Can Start and stop a VM, and view all information.
Auditors	They are the least privileged users. They have Read-only permission and therefore, can’t modify any configuration.

Benefits of RBAC

There are a number of benefits to using RBAC to restrict unnecessary network access based on people's roles within an organization, including:

- Improving operational efficiency.

  Having predefined roles in RBAC makes it is easy to include new users with the right privileges or switch roles of existing users. It also cuts down on the potential for error when user permissions are being assigned.

- Enhancing compliance.

Every organization must comply with local, state and federal regulations. Companies generally prefer to implement RBAC systems to meet the regulatory and statutory requirements for confidentiality and privacy because executives and IT departments can more effectively manage how the data is accessed and used. This is particularly important for financial institutions and healthcare companies that manage sensitive data.

- Reducing costs.

  By not allowing user access to certain processes and applications, companies may conserve or use resources such as network bandwidth, memory and storage in a cost-effective manner.

- Decreasing risk of breaches and data leakage.

  Implementing RBAC means restricting access to sensitive information, thus reducing the potential for data breaches or data leakage.

Best practices for role-based access control implementations

- As an administrator, determine the list of users and assign the users to the predefined roles. For example, the user "networkadmin" can be created and added to the user group "administrators".

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



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

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

- Terminate accounts for users who no longer require access.

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

- Periodically conduct audits to evaluate the roles, the employees who are assigned to them and the access that's permitted for each role. If a user is found to have unnecessary access to a certain system, change the user's role.

For more details see, Users, Roles, and Authentication, on page 27

Granular Role-Based Access Control

Starting from NFVIS 4.7.1, the Granular Role-Based Access Control feature is introduced. This feature adds a new resource group policy that manages the VM and VNF and allows you to assign users to a group to control VNF access, during VNF deployment. For more information, see Granular Role-Based Access Control.

Restrict Device Accessibility

Users have repeatedly been caught unawares by attacks against features they had not protected because they did not know that those features were enabled. Unused services tend to be left with default configurations which are not always secure. These services may also be using default passwords. Some services can give an attacker easy access to information on what the server is running or how the network is setup. The following sections describe how NFVIS avoids such security risks:

Attack vector reduction

Any piece of software can potentially contain security vulnerabilities. More software means more avenues for attack. Even if there are no publicly known vulnerabilities at the time of inclusion, vulnerabilities will probably be discovered or disclosed in the future. To avoid such scenarios, only those software packages which are essential for the NFVIS functionality are installed. This helps to limit software vulnerabilities, reduce resource consumption, and reduce extra work when problems are found with those packages. All third-party software included in NFVIS is registered at a central database in Cisco so that Cisco is able to perform a company level organized response (Legal, Security, etc). Software packages are periodically patched in every release for known Common Vulnerabilities and Exposures (CVEs).

Enabling only essential ports by default

Only those services which are absolutely necessary to setup and manage NFVIS are available by default. This removes the user effort needed to configure firewalls and deny access to unnecessary services. The only services that are enabled by default are listed below along with the ports they open.

Open Port	Service	Description
22/TCP	SSH	Secure Socket Shell for remote command-line access to NFVIS
80/TCP	HTTP	Hypertext Transfer Protocol for the NFVIS portal access. All HTTP traffic received by NFVIS is redirected to port 443 for HTTPS
443/TCP	HTTPS	Hypertext Transfer Protocol Secure for secure NFVIS portal access
830/TCP	NETCONF-ssh	Port opened for the Network Configuration Protocol (NETCONF) over SSH. NETCONF is a protocol used for automated configuration of NFVIS and for receiving asynchronous event notifications from NFVIS.
161/UDP	SNMP	Simple Network Management Protocol (SNMP). Used by NFVIS to communicate with remote network-monitoring applications. For more information see, Introduction about SNMP, on page 340

Restrict Access To Authorized Networks For Authorized Services

Only authorized originators should be permitted to even attempt device management access, and access should be only to the services they are authorized to use. NFVIS can be configured such that access is restricted to known, trusted sources and expected management traffic profiles. This reduces the risk of unauthorized access and the exposure to other attacks, such as brute force, dictionary, or DoS attacks.

To protect the NFVIS management interfaces from unnecessary and potentially harmful traffic, an admin user can create Access Control Lists (ACLs) for the network traffic that is received. These ACLs specify the source IP addresses/networks from which the traffic originates, and the type of traffic that is permitted or rejected from these sources. These IP traffic filters are applied to each management interface on NFVIS. The following parameters are configured in an IP receive Access Control List (ip-receive-acl)

Parameter	Value	Description
Source network/Netmask	Network/netmask. For example: 0.0.0.0/0 172.39.162.0/24	This field specifies the IP address/network from which the traffic originates
Service	https icmp netconf scpd snmp ssh	Type of traffic from the specified source.
Action	accept drop reject	Action to be taken on the traffic from the source network. With **accept** , new connection attempts will be granted. With **reject** , connection attempts will not be accepted. If the rule is for a TCP based service such as HTTPS, NETCONF, SCP, SSH, the source will get a TCP reset (RST) packet. For non-TCP rules such as SNMP and ICMP, the packet will be dropped. With **drop**, all packets will be dropped immediately, there is no information sent to the source.

Parameter	Value	Description
Priority	A numeric value	The **priority** is used to enforce an order on the rules. Rules with a higher numeric value for priority will be added further down in the chain. If you want to make sure that a rule will be added after another one, use a low priority number for the first and a higher priority number for the following.

The following sample configurations illustrate some scenarios that can be adapted for specific use-cases.

Configuring the IP Receive ACL

The more restrictive an ACL, the more limited the exposure to unauthorized access attempts. However, a more restrictive ACL can create a management overhead, and can impact accessibility to perform troubleshooting. Consequently, there is a balance to be considered. One compromise is to restrict access to internal corporate IP addresses only. Each customer must evaluate the implementation of ACLs in relation to their own security policy, risks, exposure, and acceptance thereof.

Reject ssh traffic from a subnet:

```
nfvis(config)# system settings ip-receive-acl 171.70.63.0/24 service ssh action reject
priority 1
```

Removing ACLs:

When an entry is deleted from **ip-receive-acl**, all configurations to that source are deleted since the source IP address is the key. To delete just one service, configure other services again.

```
nfvis(config)# no system settings ip-receive-acl 171.70.63.0/24
```

For more details see, Configuring the IP Receive ACL, on page 21

Privileged Debug Access

The super-user account on NFVIS is disabled by default, to prevent all unrestricted, potentially adverse, system-wide changes and NFVIS does not expose the system shell to the user.

However, for some hard to debug issues on the NFVIS system, the Cisco Technical Assistance Center team (TAC) or development team might require shell access to the customer's NFVIS. NFVIS has a secure unlock infrastructure to ensure that privileged debug access to a device in the field is restricted to authorized Cisco employees. To securely access the Linux shell for this kind of interactive debugging, a challenge-response authentication mechanism is used between NFVIS and the Interactive debugging server maintained by Cisco. The admin user's password is also required in addition to the challenge-response entry to ensure that the device is accessed with the customer's consent.

Steps to access the shell for Interactive Debugging:

1. An admin user initiates this procedure using this hidden command.

```
nfvis# system shell-access
```

2. The screen will show a challenge string, for example:

```
Challenge String (Please copy everything between the asterisk lines exclusively):

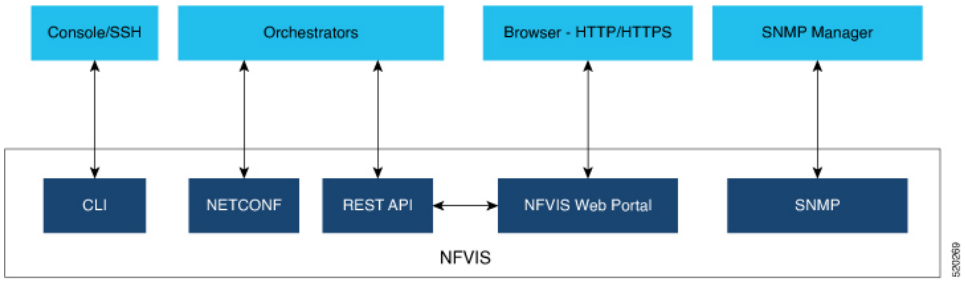
********************************************************************************
SPH//wkAAABORlZJU0VOQ1M1NDA4L0s5AQAAABt+dcx+hB0V06r9RkdMMjEzNTgw
RlHq7BxeAAA=
DONE.
********************************************************************************
```

3. The Cisco member enters the Challenge string on an Interactive Debug server maintained by Cisco. This server verifies that the Cisco user is authorized to debug NFVIS using the shell, and then returns a response string.
4. Enter the response string on the screen below this prompt:

   Input your response when ready:
5. When prompted, the customer should enter the admin password.
6. You get shell-access if the password is valid.
7. Development or TAC team uses the shell to proceed with the debugging.
8. To exit shell-access type **Exit**.

Secure Interfaces

NFVIS management access is allowed using the interfaces shown in the diagram. The following sections describe security best practices for these interfaces to NFVIS.



Console

The console port is an asynchronous serial port that allows you to connect to the NFVIS CLI for initial configuration. A user can access the console with either physical access to the NFVIS or remote access through the use of a terminal server. If console port access is required via a terminal server, configure access lists on the terminal server to allow access only from the required source addresses.

SSH

Users can access the NFVIS CLI by using SSH as a secure means of remote login. The integrity and confidentiality of NFVIS management traffic is essential to the security of the administered network since administration protocols frequently carry information which could be used to penetrate or disrupt the network.

NFVIS uses SSH version 2, which is Cisco's and the Internet's de facto standard protocol for interactive logins and supports strong encryption, hash, and key exchange algorithms recommended by the Security and Trust Organization within Cisco.

CLI Session timeout

By logging in via SSH, a user establishes a session with NFVIS. While the user is logged in, if the user leaves the logged-in session unattended, this can expose the network to a security risk. Session security limits the risk of internal attacks, such as one user trying to use another user's session.

To mitigate this risk, NFVIS times out CLI sessions after 15 minutes of inactivity. When the session timeout is reached, the user is automatically logged out.

NETCONF

The Network Configuration Protocol (NETCONF) is a Network Management protocol developed and standardized by the IETF for the automated configuration of network devices.

The NETCONF protocol uses an Extensible Markup Language (XML) based data encoding for the configuration data as well as the protocol messages. The protocol messages are exchanged on top of a secure transport protocol.

NETCONF allows NFVIS to expose an XML-based API that the network operator can use to set and get configuration data and event notifications securely over SSH.

For more information see, NETCONF Event Notifications, on page 339.

REST API

NFVIS can be configured using RESTful API over HTTPS. The REST API allow the requesting systems to access and manipulate the NFVIS configuration by using a uniform and predefined set of stateless operations. Details on all the REST APIs can be found in the NFVIS API Reference guide.

When the user issues a REST API, a session is established with NFVIS. In order to limit risks related to denial of service attacks, NFVIS limits the total number of concurrent REST sessions to 100.

NFVIS Web Portal

The NFVIS portal is a web-based Graphical User Interface that displays information about NFVIS. The portal presents the user with an easy means to configure and monitor NFVIS over HTTPS without having to know the NFVIS CLI and API.

Session Management

The stateless nature of HTTP and HTTPS requires a method of uniquely tracking users through the use of unique session IDs and cookies.

NFVIS encrypts the user's session. The AES-256-CBC cipher is used to encrypt the session contents with an HMAC-SHA-256 authentication tag. A random 128-bit Initialization Vector is generated for each encryption operation.

An Audit record is started when a portal session is created. Session information is deleted when the user logs out or when the session times out.

The default idle timeout for portal sessions is 15 minutes. However, this can be configured for the current session to a value between 5 and 60 minutes on the Settings page. Auto-logout will be initiated after this

period. Multiple sessions are not permitted in a single browser. The Maximum number of concurrent sessions are set to 30.

The NFVIS portal utilizes cookies to associate data with the user. It uses the following cookie properties for enhanced security:

- **ephemeral** to ensure the cookie expires when the browser is closed
- **httpOnly** to make the cookie inaccessible from JavaScript
- **secureProxy** to ensure the cookie can only be sent over SSL.

Even after authentication, attacks such as Cross-Site Request Forgery (CSRF) are possible. In this scenario, an end user might inadvertently execute unwanted actions on a web application in which they're currently authenticated. To prevent this, NFVIS uses **CSRF** tokens to validate every REST API that is invoked during each session.

URL Redirection

In typical web servers, when a page is not found on the web server, the user gets a 404 message; for pages that exist, they get a login page. The security impact of this is that an attacker can perform a brute force scan and easily detect which pages and folders exist.

To prevent this on NFVIS, all non-existent URLs prefixed with the device IP are redirected to the portal login page with a 301 status response code. This means that irrespective of the URL requested by an attacker, they will always get the login page to authenticate themselves.

All HTTP server requests are redirected to HTTPS and have the following headers configured:

- X-Content-Type-Options
- X-XSS-Protection
- Content-Security-Policy
- X-Frame-Options
- Strict-Transport-Security
- Cache-Control

Disabling the Portal

The NFVIS portal access is enabled by default. If you are not planning to use the portal, it is recommended to disable portal access using this command:

```
Configure terminal
System portal access disabled
commit
```

HTTPS

All the HTTPS data to and from NFVIS uses Transport Layer Security (TLS) to communicate across the network. TLS is the successor to Secure Socket Layer (SSL).

The TLS handshake involves authentication during which the client verifies the server's SSL certificate with the certificate authority that issued it. This confirms that the server is who it says it is, and that the client is interacting with the owner of the domain.

By default, NFVIS uses a self-signed certificate to prove its identity to its clients. This certificate has a 2048-bit public key to increase the security of the TLS encryption, since the encryption strength is directly related to the key size.

Certificate Management

NFVIS generates a self-signed SSL certificate when first installed. It is a security best practice to replace this certificate with a valid certificate signed by a compliant Certificate Authority (CA).

Use the following steps to replace the default self-signed certificate:

1. Generate a Certificate Signing Request (CSR) on NFVIS.

   A Certificate Signing request (CSR) is a file with a block of encoded text that is given to a Certificate Authority when applying for an SSL Certificate. This file contains information that should be included in the certificate such as the organization name, common name (domain name), locality, and country. The file also contains the public key that should be included in the certificate. NFVIS uses a 2048-bit public key since encryption strength is higher with a higher key size.

   To generate a CSR on NFVIS, run the following command:

   ```
   nfvis# system certificate signing-request [common-name  country-code  locality
   organization  organization-unit-name  state]
   ```

   The CSR file is saved as */data/intdatastore/download/nfvis.csr*.

   .

2. Get an SSL certificate from a CA using the CSR.

   From an external host, use the scp command to download the Certificate Signing Request.

   ```
   [myhost:/tmp] > scp -P 22222 admin@<NFVIS-IP>:/data/intdatastore/download/nfvis.csr
   <destination-file-name>
   ```

   Contact a Certificate authority to issue a new SSL server certificate using this CSR.

3. Install the CA Signed Certificate.

   From an external server, use the scp command to upload the certificate file into NFVIS to the *data/intdatastore/uploads/* directory.

   ```
   [myhost:/tmp] > scp -P 22222 <certificate file>
   admin@<NFVIS-IP>:/data/intdatastore/uploads
   ```

   Install the certificate in NFVIS using the following command.

   ```
   nfvis# system certificate install-cert path file:///data/intdatastore/uploads/<certificate
    file>
   ```

4. Switch to using the CA Signed Certificate.

   Use the following command to start using the CA signed certificate instead of the default self-signed certificate.

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

SNMP Access

Simple Network Management Protocol (SNMP) is an Internet Standard protocol for collecting and organizing information about managed devices on IP networks, and for modifying that information to change device behavior.

Three significant versions of SNMP have been developed. NFVIS supports SNMP version 1, version 2c and version 3. SNMP versions 1 and 2 use community strings for authentication, and these are sent in plain-text. So, it is a security best practice to use SNMP v3 instead.

SNMPv3 provides secure access to devices by using three aspects: - users, authentication, and encryption. SNMPv3 uses the USM (User-based Security Module) for controlling access to information available via SNMP. The SNMP v3 user is configured with an authentication type, a privacy type as well as a passphrase. All users sharing a group utilize the same SNMP version, however, the specific security level settings (password, encryption type, etc.) are specified per-user.

The following table summarizes the security options within SNMP

Model	Level	Authentication	Encyption	Outcome
v1	noAuthNoPriv	Community String	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community String	No	Uses a community string match for authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the HMAC-MD5-96 or HMAC-SHA-96 algorithms.

Model	Level	Authentication	Encyption	Outcome
v3	authPriv	MD5 or SHA	Data Encryption Standard (DES) or Advanced Encryption Standard (AES)	Provides authentication based on the HMAC-MD5-96 or HMAC-SHA-96 algorithms. Provides DES Cipher algorithm in Cipher Block Chaining Mode (CBC-DES) or AES encryption algorithm used in Cipher FeedBack Mode (CFB), with a 128-bit key size(CFB128-AES-128)

Since its adoption by NIST, AES has become the dominant encryption algorithm throughout the industry. To follow the industry's migration away from MD5 and toward SHA, it is a security best practice to configure the SNMP v3 authentication protocol as SHA and privacy protocol as AES.

For more details on SNMP see, Introduction about SNMP, on page 340

Legal Notification Banners

It is recommended that a legal notification banner is present on all interactive sessions to ensure that users are notified of the security policy being enforced and to which they are subject. In some jurisdictions, civil and/or criminal prosecution of an attacker who breaks into a system is easier, or even required, if a legal notification banner is presented, informing unauthorized users that their use is in fact unauthorized. In some jurisdictions, it may also be forbidden to monitor the activity of an unauthorized user unless they have been notified of the intent to do so.

Legal notification requirements are complex and vary in each jurisdiction and situation. Even within jurisdictions, legal opinions vary. Discuss this issue with your own legal counsel to ensure that the notification banner meets company, local, and international legal requirements. This is often critical to securing appropriate action in the event of a security breach. In cooperation with the company legal counsel, statements which may be included in a legal notification banner include:

- Notification that the system access and use is permitted only by specifically authorized personnel, and perhaps information about who may authorize use.
- Notification that unauthorized access and use of the system is unlawful, and may be subject to civil and/or criminal penalties.
- Notification that access and use of the system may be logged or monitored without further notice, and the resulting logs may be used as evidence in court.
- Additional specific notices required by specific local laws.

From a security rather than a legal point of view, a legal notification banner should not contain any specific information about the device, such as its name, model, software, location, operator or owner because this kind of information may be useful to an attacker.

The following is a sample legal notification banner which can be displayed before login:

```
UNAUTHORIZED ACCESS TO THIS DEVICE IS PROHIBITED You must have explicit, authorized permission
 to access or configure this device. Unauthorized attempts and actions to access or use
this system may result in civil and/or criminal penalties. All activities performed on this
 device are logged and monitored
```



Note Present a legal notification banner approved by company legal counsel.

NFVIS allows the configuration of a banner and Message of the Day (MOTD). The banner is displayed before the user logs in. Once the user logs in to NFVIS, a system-defined banner provides Copyright information about NFVIS, and the message-of-the-day (MOTD), if configured, will appear, followed by the command line prompt or portal view, depending on the login method.

It is recommended that a login banner is implemented to ensure that a legal notification banner is presented on all the device management access sessions prior to a login prompt being presented. Use this command to configure the banner and MOTD.

```
nfvis(config)# banner-motd banner <banner-text> motd <message-of-the-day-text>
```

For more information about the banner command, see Configure Banner, Message of the day and System Time, on page 85.

Factory Default Reset

Factory Reset removes all the customer specific data that has been added to the device since the time of its shipping. The data erased includes configurations, log files, VM images, connectivity information, and user login credentials.

It provides one command to reset the device to factory-original settings, and is useful in the following scenarios:

- Return Material Authorization (RMA) for a device—If you have to return a device to Cisco for RMA, use Factory Default reset to remove all the customer-specific data.
- Recovering a compromised device— If the key material or credentials stored on a device is compromised, reset the device to factory configuration and then reconfigure the device.
- If the same device needs to be re-used at a different site with a new configuration, perform a Factory Default reset to remove the existing configuration and bring it to a clean state.

NFVIS provides the following options within Factory default reset:

Factory Reset Option	Data Erased	Data Retained
all	All configuration, uploaded image files, VMs and logs. Connectivity to the device will be lost.	The admin account is retained and the password will be changed to the factory default password.

Factory Reset Option	Data Erased	Data Retained
all-except-images	All configuration except image configuration, VMs, and uploaded image files. Connectivity to the device will be lost.	Image configuration, registered images and logs The admin account is retained and the password will be changed to the factory default password.
all-except-images-connectivity	All configuration except image, network and connectivity configuration, VMs, and uploaded image files. Connectivity to the device is available.	Images, network and connectivity related configuration, registered images, and logs. The admin account is retained and the previously configured admin password will be preserved.
manufacturing	All configuration except image configuration, VMs, uploaded image files, and logs. Connectivity to the device will be lost.	Image related configuration and registered images The admin account is retained and the password will be changed to the factory default password.

The user must choose the appropriate option carefully based on the purpose of the Factory Default reset.

For more information, see Resetting to Factory Default, on page 84.

Infrastructure Management Network

An infrastructure management network refers to the network carrying the control and management plane traffic (such as NTP, SSH, SNMP, syslog, etc.) for the infrastructure devices. Device access can be through the console, as well as through the Ethernet interfaces. This control and management plane traffic is critical to network operations, providing visibility into and control over the network. Consequently, a well-designed and secure infrastructure management network is critical to the overall security and operations of a network. One of the key recommendations for a secure infrastructure management network is the separation of management and data traffic in order to ensure remote manageability even under high load and high traffic conditions. This can be achieved using a dedicated management interface.

The following are the Infrastructure management network implementation approaches:

Out-of-band Management

An Out-of-band Management (OOB) management network consists of a network which is completely independent and physically disparate from the data network that it helps to manage. This is also sometimes referred to as a Data Communications Network (DCN). Network devices can connect to the OOB network in different ways: – NFVIS supports a built-in management interface that can be used to connect to the OOB network. NFVIS allows the configuration of a predefined physical interface, the MGMT port on the ENCS, as a dedicated management interface. Restricting management packets to designated interfaces provides greater control over the management of a device, thereby providing more security for that device. Other benefits include improved performance for data packets on non-management interfaces, support for network scalability,

need for fewer access control lists (ACLs) to restrict access to a device, and prevention of management packet floods from reaching the CPU.

Network devices can also connect to the OOB network via dedicated data interfaces. In this case, ACLs should be deployed to ensure that management traffic is only handled by the dedicated interfaces.

For further information, see Configuring the IP Receive ACL, on page 21 and Port 22222 and Management Interface ACL, on page 21.

Pseudo out-of-band Management

A pseudo out-of-band management network uses the same physical infrastructure as the data network but provides logical separation through the virtual separation of traffic, by using VLANs. NFVIS supports creating VLANs and virtual bridges to help identify different sources of traffic and separate traffic between VMs. Having separate bridges and VLANs isolates the virtual machine network's data traffic and the management network, thus providing traffic segmentation between the VMs and the host. For further information see Configuring VLAN for NFVIS Management Traffic, on page 21.

In-band Management

An in-band management network uses the same physical and logical paths as the data traffic.

Ultimately, this network design requires a per-customer analysis of risk versus benefits and costs. Some general considerations include:

- An isolated OOB management network maximizes visibility and control over the network even during disruptive events.
- Transmitting network telemetry over an OOB network minimizes the chance for disruption of the very information which provides critical network visibility.
- In-band management access to network infrastructure, hosts, etc. is vulnerable to complete loss in the event of a network incident, removing all the network visibility and control. Appropriate QoS controls should be put in place to mitigate this occurrence.
- NFVIS features interfaces which are dedicated to device management, including serial console ports and Ethernet management interfaces.
- An OOB management network can typically be deployed at a reasonable cost, since management network traffic does not typically demand high bandwidth nor high performance devices, and only requires sufficient port density to support the connectivity to each infrastructure device.

Locally Stored Information Protection

Protecting Sensitive Information

NFVIS stores some sensitive information locally, including passwords and secrets. Passwords should generally be maintained and controlled by a centralized AAA server. However, even if a centralized AAA server is deployed, some locally-stored passwords are required for certain cases such as local fallback in the case of AAA servers not being available, special-use usernames, etc. These local passwords and other sensitive

information are stored on NFVIS as hashes so that it is not possible to recover the original credentials from the system. Hashing is a widely accepted industry norm.

File Transfer

Files which may need to be transferred to NFVIS devices include VM image and NFVIS upgrade files. The secure transfer of files is critical for network infrastructure security. NFVIS supports Secure Copy (SCP) to ensure the security of file transfer. SCP relies on SSH for secure authentication and transport, enabling the secure and authenticated copying of files.

A secure copy from NFVIS is initiated through the scp command. The secure copy (scp) command allows only the admin user to securely copy files from NFVIS to an external system, or from an external system to NFVIS.

The syntax for the scp command is:

```
scp <source> <destination>
```

We use port 22222 for the NFVIS SCP server. By default, this port is closed and users cannot secure copy files into NFVIS from an external client. If there is a need to SCP a file from an external client, the user can open the port using:

```
system settings ip-receive-acl (address)/(mask lenth)  service scpd priority (number) action
 accept
commit
```

To prevent users from accessing system directories, secure copy can be performed only to or from intdatastore:, extdatastore1:, extdatastore2:, usb: and nfs:, if available. Secure copy can also be performed from logs: and techsupport:

Logging

NFVIS access and configuration changes are logged as audit logs to record the following information:

- Who accessed the device
- When did a user log in
- What did a user do in terms of the host configuration and the VM lifecycle
- When did a user log off
- Failed access attempts
- Failed authentication requests
- Failed authorization requests

This information is invaluable for forensic analysis in case of unauthorized attempts or access, as well as for configuration change issues and to help plan group administration changes. It may also be used real time to identify anomalous activities which may indicate that an attack is taking place. This analysis can be correlated with information from additional external sources, such as IDS and firewall logs.

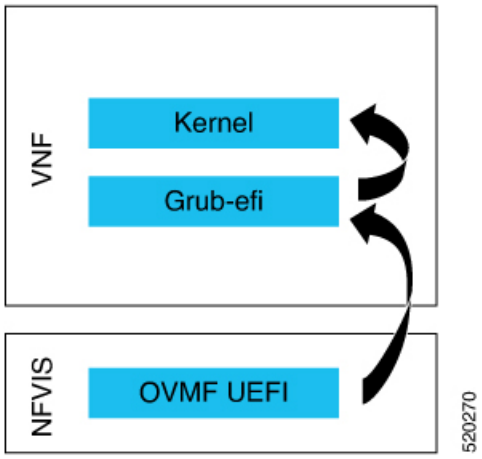
All the key events on the NFVIS are sent as event notifications to NETCONF subscribers and as syslogs to the configured central logging servers. For more information on syslog messages and event notifications, see Appendix, on page 375.

Virtual Machine security

This section describes security features related to the registration, deployment and operation of Virtual Machines on NFVIS.

VNF secure boot

NFVIS supports Open Virtual Machine Firmware (OVMF) to enable UEFI secure boot for Virtual Machines which support secure boot. VNF Secure boot verifies that each layer of the VM boot software is signed, including the bootloader, the operating system kernel, and operating system drivers.



For more information see, Secure Boot of VNFs, on page 162.

VNC Console Access Protection

NFVIS allows the user to create a Virtual Network Computing (VNC) session to access a deployed VM's remote desktop. To enable this, NFVIS dynamically opens a port to which the user can connect using their web browser. This port is only left open for 60 seconds for an external server to start a session to the VM. If no activity is seen within this time, the port is closed. The port number is assigned dynamically and thereby allows only a one-time access to the VNC console.

```
nfvis# vncconsole start deployment-name 1510614035 vm-name ROUTER
vncconsole-url :6005/vnc_auto.html
```

Pointing your browser to https://<nfvis ip>:6005/vnc_auto.html will connect to the ROUTER VM's VNC console.

Encrypted VM config data variables

During VM deployment, the user provides a day-0 configuration file for the VM. This file can contain sensitive information such as passwords and keys. If this information is passed as clear text, it appears in log files and internal database records in clear text. This feature allows the user to flag a config data variable as sensitive so that its value is encrypted using AES-CFB-128 encryption before it is stored or passed to internal subsystems.

For more information see, VM Deployment Parameters, on page 135.

Checksum verification for Remote Image Registration

To register a remotely located VNF image, the user specifies its location. The image will need to be downloaded from an external source, such as an NFS server or a remote HTTPS server.

To know if a downloaded file is safe to install, it is essential to compare the file's checksum before using it. Verifying the checksum helps ensure that the file was not corrupted during network transmission, or modified by a malicious third party before you downloaded it.

NFVIS supports the checksum and checksum_algorithm options for the user to provide the expected checksum and checksum algorithm (SHA256 or SHA512) to be used to verify the checksum of the downloaded image. Image creation fails if the checksum does not match.

Certification Validation for Remote Image Registration

To register a VNF image located on a HTTPS server, the image will need to be downloaded from the remote HTTPS server. To securely download this image, NFVIS verifies the SSL certificate of the server. The user needs to specifying either the path to the certificate file or the PEM format certificate contents to enable this secure download.

More details can be found at Section on certificate validation for image registration

VM Isolation and Resource provisioning

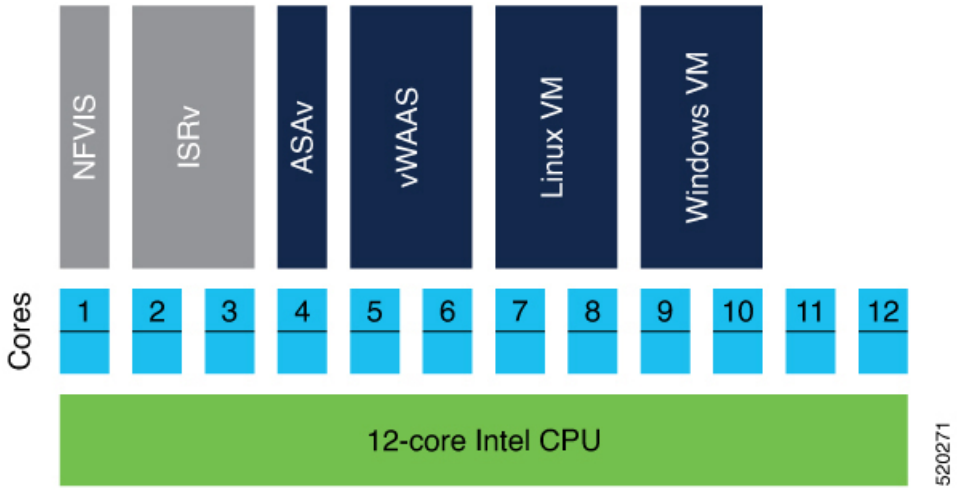
The Network Function Virtualization (NFV) architecture consists of:

- Virtualized network functions (VNFs), which are Virtual Machines running software applications that deliver network functionality such as a router, firewall, load balancer, and so on.
- Network functions virtualization infrastructure, which consists of the infrastructure components—compute, memory, storage, and networking, on a platform that supports the required software and hypervisor.

With NFV, network functions are virtualized so that multiple functions can be run on a single server. As a result, less physical hardware is needed, allowing for resource consolidation. In this environment, it is essential to simulate dedicated resources for multiple VNFs from a single, physical hardware system. Using NFVIS, VMs can be deployed in a controlled manner such that each VM receives the resources it needs. Resources are partitioned as needed from the physical environment to the many virtual environments. The individual VM domains are isolated so they are separate, distinct, and secure environments, which are not contending with each other for shared resources.

VMs cannot use more resources than provisioned. This avoids a Denial of Service condition from one VM consuming the resources. As a result, CPU, memory, network and storage are protected.

CPU Isolation



The NFVIS system reserves cores for the infrastructure software running on the host. The rest of the cores are available for VM deployment. This guarantees that the VM's performance does not affect the NFVIS host performance.

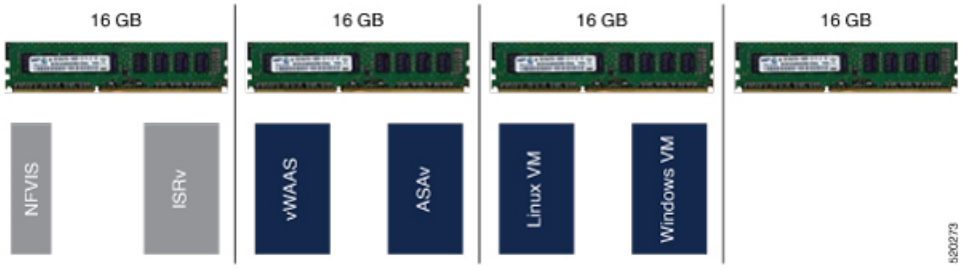
Low-latency VMs

NFVIS explicitly assigns dedicated cores to low latency VMs that are deployed on it. If the VM requires 2 vCPUs, it is assigned 2 dedicated cores. This prevents sharing and oversubscription of cores and guarantees the performance of the low-latency VMs. If the number of available cores is less than the number of vCPUs requested by another low-latency VM, the deployment is prevented since we do not have sufficient resources.

Non low-latency VMs

NFVIS assigns sharable CPUs to non low latency VMs. If the VM requires 2 vCPUs, it is assigned 2 CPUs. These 2 CPUs are shareable among other non low latency VMs. If the number of available CPUs is less than the number of vCPUs requested by another non low-latency VM, the deployment is still allowed because this VM will share the CPU with existing non low latency VMs.

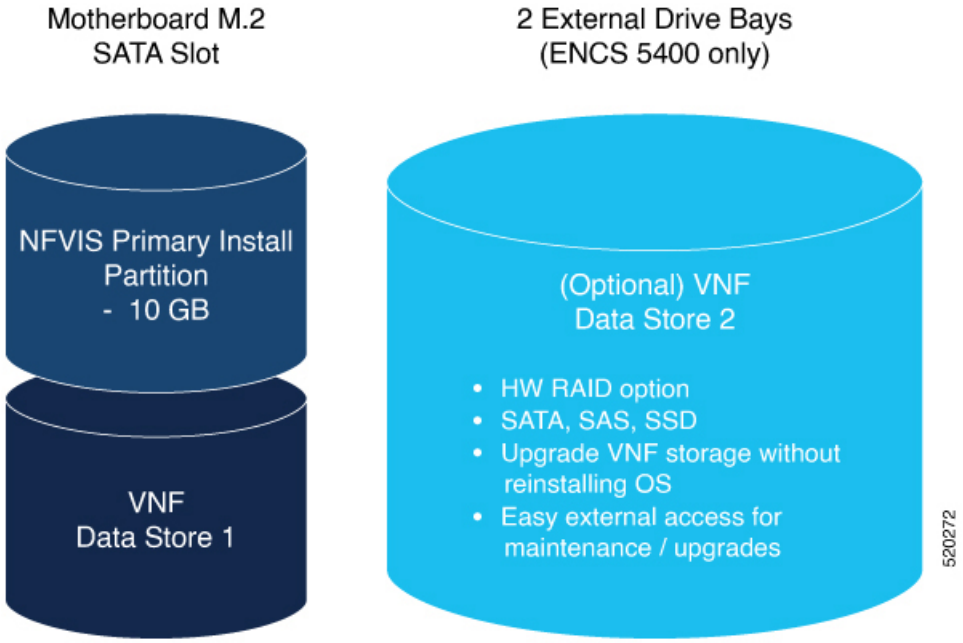
Memory Allocation



The NFVIS Infrastructure requires a certain amount of memory. When a VM is deployed, there is a check to ensure that the memory available after reserving the memory required for the infrastructure and previously deployed VMs, is sufficient for the new VM. We do not allow memory oversubscription for the VMs.

VMs are not allowed to directly access the host file system and storage.

Storage Isolation



The ENCS platform supports an internal datastore (M2 SSD) and external disks. NFVIS is installed on the internal datastore. VNFs can also be deployed on this internal datastore. It is a security best practice to store customer data and deploy customer application Virtual Machines on the external disks. Having physically separate disks for the system files vs the application files helps to protect system data from corruption and security issues.

-

Interface Isolation



Single Root I/O Virtualization or SR-IOV is a specification that allows the isolation of PCI Express (PCIe) resources such as an Ethernet port. Using SR-IOV a single Ethernet port can be made to appear as multiple, separate, physical devices known as Virtual Functions. All of the VF devices on that adapter share the same physical network port. A guest can use one or more of these Virtual Functions. A Virtual Function appears to the guest as a network card, in the same way as a normal network card would appear to an operating system.

Virtual Functions have near-native performance and provide better performance than para-virtualized drivers and emulated access. Virtual Functions provide data protection between guests on the same physical server as the data is managed and controlled by the hardware.

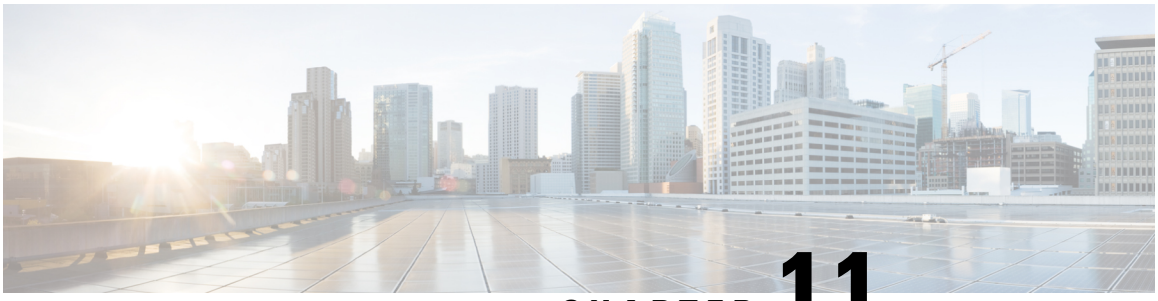
NFVIS VNFs can use SR-IOV networks to connect to WAN and LAN Backplane ports.

Each such VM owns a virtual interface and its related resources achieving data protection among VMs.

Secure Development Lifecycle

NFVIS follows a Secure Development Lifecycle (SDL) for software. This is a repeatable, measurable process designed to reduce vulnerabilities and enhance the security and resilience of Cisco solutions. Cisco SDL applies industry-leading practices and technology to build trustworthy solutions that have fewer field-discovered product security incidents. Every NFVIS release goes through the following processes.

- Following Cisco-internal and market-based Product Security Requirements
- Registering 3rd party software with a central repository at Cisco for vulnerability tracking
- Periodically patching software with known fixes for CVEs.
- Designing software with Security in mind
- Following secure coding practices such as using vetted common security modules like CiscoSSL, running Static Analysis and implementing input validation for Preventing command injection, etc.
- Using Application Security tools such as IBM AppScan, Nessus, and other Cisco internal tools.



CHAPTER 11

Secure Operation in FIPS Mode on NFVIS

Table 49: Feature History

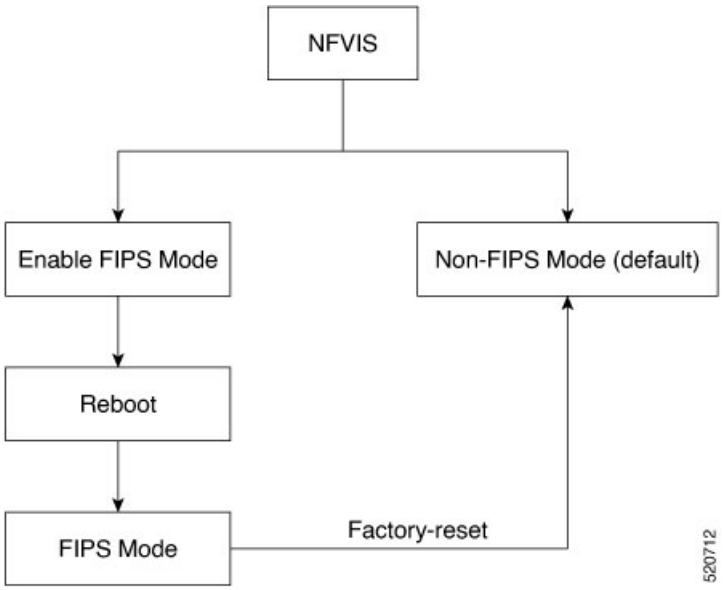
Feature Name	Release Information	Description
Secure Operation in FIPS Mode on NFVIS	NFVIS 4.2.1	The Federal Information Processing Standards (FIPS) Publication 140-2 are publicly announced standards developed by the United States federal government for use in computer systems by non-military government agencies and government contractors.

The Federal Information Processing Standards (FIPS) Publication 140-2 are publicly announced standards developed by the United States federal government for use in computer systems by non-military government agencies and government contractors. In FIPS mode, the device tries to prevent the use of non-FIPS compatible algorithms, but you must ensure that you configure the device to use only FIPS approved algorithms. Some functionality may silently fail in FIPS mode if it attempts to use non-FIPS compliant algorithms.

- Configuring FIPS Mode, on page 253
- FIPS Operational Status, on page 256

Configuring FIPS Mode

NFVIS supports both FIPS and non-FIPS mode of operation. NFVIS is FIPS 140-2 compliant for SSH, TLS and SNMP protocols.



To enable FIPS mode:

```
config terminal
security fips
commit
```

Only after NFVIS reboot after the configuration FIPS mode is enabled.

FIPS mode can be disabled only with factory reset. If you try to disable FIPS mode with the no form of the command:

```
config terminal
no security fips
commit
Aborted: This command can only be removed while factory reset
```

The following is an example, where FIPS mode is successfully configured, but not enabled:

```
nfvis# show security
security fips-status CONFIGURED_REBOOT_TO_ENABLE
```

To verify the status of the FIPS mode after reboot:

```
nfvis# show security
security fips-status ENABLED
```

FIPS mode configuration is terminated when:

- SNMP v1 or v2 is configured.

  The following is an example of FIPS mode configuration failure when SNMP v1 or v2 is configured:

  ```
  config terminal
  security fips
  commit
  Aborted: SNMP version 1 and/or SNMP version 2 is configure. Please unconfigure SNMPv1
  and SNMPv2 and then try again
  ```

- SNMP v3 is configured with auth protocol MD5.

  The following is an example of FIPS mode configuration failure when SNMP v3 is configured with auth protocol md5:

```
config terminal
security fips
commit
Aborted: SNMP version 3 MD5 auth-protocol configured other secure protocol and try again
```



Note After FIPS mode is enabled, SNMP v1or v2 and SNMP v3 with auth protocol MD5 cannot be configured.

```
snmp group test_v1 snmp 1 noAuthNoPriv read test write test
commit
Aborted: Cannot configure SNMP group-version 1 because fips-status is ENABLED
```

```
config terminal
snmp user test_md5_v3
    user-version  3
    user-group test_v3
    auth-protocol md5
    auth-key 46:97:c3:b0:ba:45:fd:5e:be:99:44:c5:64:c9:bc:44
    commit
Aborted: 'snmp user test_md5_v3_passhd auth-protocol': Cannot configure SNMP user-version
3 with auth-protocol MD5 because fips-status is CONFIGURED_REBOOT_TO_ENABLE
nfvis(config-user-test_md5_v3_passhd)#
```

Backup and Restore for FIPS Mode

If you backup NFVIS configurations when FIPS mode is enabled, then upon restore, FIPS mode is configured but needs a manual reboot to enable FIPS mode.

```
Backup configuration
nfvis#
nfvis# show running-config security
security fips
nfvis# show security
security fips-status ENABLED
nfvis#

After restore
nfvis#
nfvis# show running-config security
security fips
nfvis# show security fips-status
security fips-status CONFIGURED_REBOOT_TO_ENABLE
nfvis#

After reboot
nfvis#
nfvis# show running-config security
security fips
nfvis# show security
security fips-status ENABLED
nfvis#
```

When you backup NFVIS configurations with FIPS mode disabled, but the system where you restore the configurations has FIPS mode enabled, upon restore, the NFVIS configurations disable FIPS mode but the system has to reboot for FIPS mode to be in DISABLED state.

Backup configurations

```
nfvis# show running-config security fips
% No entries found.
nfvis# show security fips-status
security fips-status DISABLED
nfvis#
```

Restore system configurations

```
nfvis#
nfvis# show running-config security
security fips
nfvis# show security
security fips-status ENABLED
nfvis#
```

After restore

```
nfvis# show running-config security
% No entries found.
nfvis# show security
security fips-status UNCONFIGURED_REBOOT_TO_DISABLE
nfvis#
```

After reboot

```
nfvis# show running-config security fips
% No entries found.
nfvis# show security fips-status
security fips-status DISABLED
nfvis#
```

FIPS Operational Status

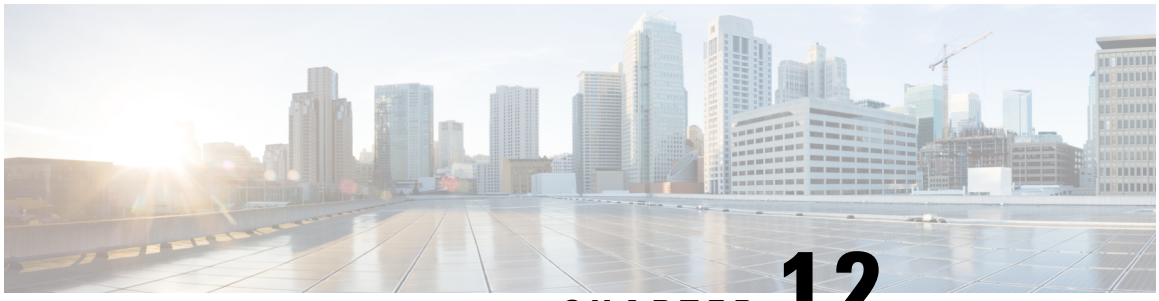
The following is a list of operational states when you try to enable FIPS mode:

- DISABLED
- CONFIGURED-REBOOT-TO-ENABLE
- ENABLED
- UNCONFIGURED-REBOOT-TO-DISABLE
- FAILED

The following table lists the possible operational state transitions for FIPS mode:

From	To	Description
DISABLED	CONFIGURED-REBOOT-TO-ENABLE	If the Oper data of fips-state leafs was previously set to DISABLED and if the **security fips** configuration is pushed
DISABLED	FAILED	If there is an error while pushing the **security fips** configuration

From	To	Description
CONFIGURED-REBOOT-TO-ENABLE	ENABLED	If the fips-mode configuration is successful before and the Oper data was set to CONFIGURED-REBOOT-TO-ENABLE, then after reboot set the Oper data to ENABLED
CONFIGURED-REBOOT-TO-ENABLE	FAILED	If the Oper data of fips-state leafs was previously set to CONFIGURED-REBOOT-TO-ENABLE and there was an error while removing fips-mode configuration
CONFIGURED-REBOOT-TO-ENABLE	DISABLED	If the fips-mode is unconfigured while restoring from a backup package or factory-reset and the current fips-status is CONFIGURED-REBOOT-TO-ENABLE
ENABLED	DISABLED	After factory-reset (of any type)
ENABLED	FAILED	If there is an error while disabling the FIPS mode
ENABLED	UNCONFIGURED-REBOOT-TO-DISABLE	If the fips-mode is unconfigured while restoring from a backup package and the current fips-status is ENABLED
FAILED	CONFIGURED-REBOOT-TO-ENABLE	If the Oper date of fips-state leafs was previously set to FAILED and now configuring fips-mode
FAILED	DISABLED	If the Oper date of fips-state leafs was previously set to FAILED and now issued factory-reset
UNCONFIGURED-REBOOT-TO-DISABLE	DISABLED	If the Oper data of fips-state leafs was previously set to UNCONFIGURED-REBOOT-TO-DISABLE and then NFVIS is rebooted
UNCONFIGURED-REBOOT-TO-DISABLE	CONFIGURED-REBOOT-TO-ENABLE	If the Oper data of fips-state leafs was previously set to UNCONFIGURED-REBOOT-TO-DISABLE and fips-mode config is successful
UNCONFIGURED-REBOOT-TO-DISABLE	FAILED	If the Oper data of fips-state leafs was previously set to UNCONFIGURED-REBOOT-TO-DISABLE and if the fips-mode config was unsuccessful



CHAPTER 12

NFVIS Clustering Capability for Cisco Cloud Services Platform

Table 50: Feature History

Feature Name	Release Information	Description
NFVIS Clustering Capability for Cisco CSP	Cisco NFVIS 4.8.1 **Note** This capability is a beta feature in this release and should not be used in production because, Cisco may make changes in the subsequent releases without providing any backward compatibility. The beta is provided as-is and without any warranty of any kind.	This feature enables you to combine three nodes into a single cluster definition so that, all the member nodes display configuration information about all the virtual machines deployed in the cluster. Each member node contributes to a user-specified disk size (same value for all the members) of their total disk size to form a shared and clustered file system. The data is replicated across all the cluster members. This clustering capability is resilient against single node failures.

- Prerequisites for NFVIS Clustering Capability for Cisco CSP, on page 259
- Restrictions for NFVIS Clustering Capability on Cisco CSP, on page 260
- Information About NFVIS Clustering Capability for Cisco CSP, on page 260
- Configure NFVIS Clustering Capability on Cisco CSP, on page 261
- Cluster File Operations, on page 264
- Support Commands for NFVIS Clustering Capability , on page 265
- Verify NFVIS Clustering Capability, on page 265
- NFVIS Clustering Capability Notifications and Syslogs, on page 267

Prerequisites for NFVIS Clustering Capability for Cisco CSP

Ensure that you configure NTP, so that the system time on all nodes of the cluster is synchronized.

Restrictions for NFVIS Clustering Capability on Cisco CSP

- To create a cluster for storage virtualization, you can use only three Cisco CSP devices with the same configuration. Configure an IP address on the new bridge. The IP addresses should match one of the three IP addresses used for creating the cluster. We recommend that you configure the 1x10 GigE NIC for data traffic. and Port channel or physical network interface card pNICs.
- When a cluster is created, the Maximum Transmission Unit (MTU) on the cluster bridge is automatically set to 9000. Ensure that the switch present in the network that connects to the PNIC supports an MTU of 9000.
- A cluster cannot be created on NFVIS hosts that have virtual machines deployed on them. If a system has virtual machine deployments, export and delete the virtual machine, before creating the cluster.
- The backup from an old host cannot be restored once the cluster is created, and the backup for a new host cannot be created after a cluster is created.
- When a cluster is created, all the nodes in the cluster share their storage with each other. Other NFVIS configurations such as additional bridges, OVS networks, users, groups, TACACS/RADIUS, and SNMP are not synchronized and must be created on each cluster node, separately.
- This feature does not support storage migration for virtual machines. Cold virtual machine migration between datastores (intdatastore, extdatastore1/2, and shared datastore) is not supported.
- You cannot add, delete, or replace an existing cluster with a new NFVIS host. If you require a new NFVIS host, in case of faulty hardware, delete the exisiting cluster and create a new cluster with the new host.

Information About NFVIS Clustering Capability for Cisco CSP

NFVIS Clustering Capability

When a device in the network is down, the virtual machines running on the device also go down and become inaccessible. Before Cisco NFVIS 4.8.1, there are two ways to recover the virtual machines:

- Recover the virtual machine from a back-up. The drawback of this method is that the backup could be outdated depending on when the backup was taken.
- Deploy the virtual machine in an active-standby mode. The drawback of this method is when the virtual machine is down, the standby takes over. However, this would require two separate licences, one for the active virtual machine and one for the standby virtual machine.

Starting from Cisco NFVIS 4.8.1, the NFVIS clustering capability is introduced. This feature allows you to form a cluster of 3 nodes. The disks of all virtual machines deployed on the cluster are replicated on all the cluster nodes. This enables you to perform a cold migration of one virtual machine to a different node in the cluster when the virtual machine's node goes down.

Guidelines to Enable Cluster Data Traffic

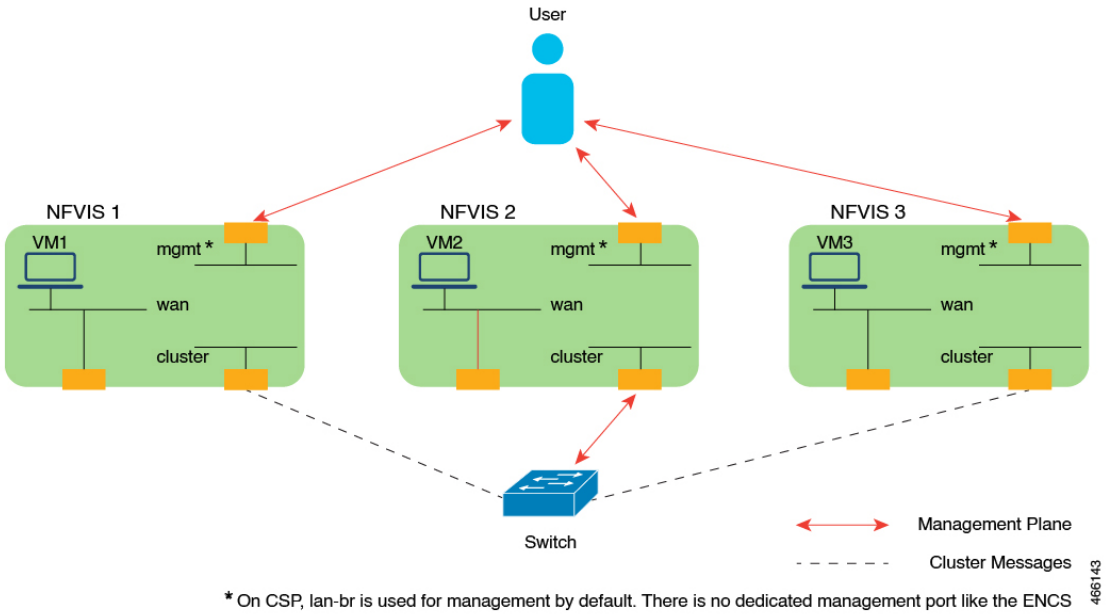
Use a dedicated pNIC to enable the flow of cluster data traffic between the nodes, and this pNIC port should be attached to a newly created bridge for the cluster traffic to flow. Configure an IP address on the new bridge. The same IP address is one of the three IP addresses used for creating the cluster.

We recommend that you configure the 1×10 GigE NIC for data traffic.

Enable Cluster Data Traffic

The following diagram illustrates a dedicated pNIC that enables the flow of cluster data traffic between the nodes. The dedicated pNIC port is attached to a newly created bridge for the cluster traffic to flow.

Figure 6: Cluster Data Traffic Flow



Configure NFVIS Clustering Capability on Cisco CSP

Before creating the cluster, ensure to complete the following steps on all the participating nodes:

1. Configure the time server.

```
nfvis# config terminal
Entering configuration mode terminal
nfvis(config)# system time ntp preferred_server YOUR-NTPD-IP
nfvis(config)# end
Uncommitted changes found, commit them? [yes/no/CANCEL] yes
Commit complete.
nfvis#
```

2. Configure the cluster bridge with IP.

```
nfvis# config t
Entering configuration mode terminal
nfvis(config)# bridges bridge cluster-br
nfvis(config-bridge-cluster-br)# ip address 198.51.100.1 255.255.255.0
nfvis(config-bridge-cluster-br)# port eth1-1
nfvis(config-port-eth1-1)# end
Uncommitted changes found, commit them? [yes/no/CANCEL] yes
Commit complete.
nfvis#
```

Note Ensure that the chosen PNIC and IP can communicate with each other.

Create a Cluster

To create a cluster:

1. Send the create cluster API call to all the Cisco NFVIS that must be added to a cluster for storage virtualization.

Note

- A cluster for storage virtualization can contain only three devices.
- The clustering API calls should be issued to all the three nodes within three minutes.
- The API call contains the name of the cluster, the IPv4 address of the nodes to be added to the cluster, and the disk size to be contributed for clustered storage. The API call also has the datastore parameter to indicate which datastore (internal or external datastore) contributes towards the storage pool.

2. After receiving the call, each server validates whether sufficient disk space is allocated for storage clustering. The server also ensures that the space utilization of the datastore is less than 80% . After the disk size validation, a reply is sent indicating that the server has received the API call. When the cluster formation is successful, a notification is pushed from the leader node to indicate that storage virtualization is complete.

The following example shows how to create a cluster:

```
nfvis# config terminal
Entering configuration mode terminal
nfvis(config)# cluster cluster1 datastore intdatastore size 10
nfvis(config-cluster-1)# node 209.165.201.1 address-type ipv4
nfvis(config-node-209.165.201.1)# exit
nfvis(config-cluster-2)# node 209.165.201.2 address-type ipv4
nfvis(config-node-209.165.201.2)# exit
nfvis(config-cluster-3)#node 209.165.201.3 address-type ipv4
nfvis(config-node-209.165.201.3)# commit
```

The cluster distributed file system is created, and it serves as a storage location for virtual machine registration and deployment. For information on verifying cluster creation, see Verify Storage Virtualization.

Register and Deploy the VM

To register and deploy VMs on the cluster, specify **gluster** as the placement option. The VM's disk details are replicated across all the other nodes on the cluster.

The following information is available to all the nodes and can be used while migrating the VM:

- The running configuration of image registration
- Flavor
- Deployment

The following parameters should be unique across the cluster (for both cluster VMs and local VMs):

- Flavor name
- Deployment names
- SRIOV networks
- int-mgmt IPs
- Port numbers used for port forwarding

During VM registration and deployment, if any of the above mentioned parameters aren't unique, the deployment is rejected.

The following example shows how to register the image:

```
nfvis(config)# vm_lifecycle images image centos src
file:///data/intdatastore/uploads/Centos_7.tar.gz properties property placement value gluster


nfvis(config-property-placement)# commit

Commit complete.
```

The following example shows how to deploy the VM:

```
nfvis(config)# vm_lifecycle tenants tenant admin deployments deployment centosvm vm_group
dep1 image centos flavor centos-small bootup_time -1 vim_vm_name centosvm placement zone_host
 host gluster
nfvis(config-placement-zone_host)# commit

Commit complete.
```

Migrate the Virtual Machine

virtual machines deployed on the cluster can be migrated to a different node on the cluster, either after they are shut down, or while the node on which they were deployed is down. Specify the API for the source and destination nodes for migration. You can choose to migrate all virtual machines from a source node or specify a list of virtual machines to migrate.

The virtual machine migration API is sent to any of the working nodes. If the host is up during migration, stop the virtual machine.

The virtual machine disk is replicated in the destination node, the virtual machine comes up in the Day-N state. After the migration, the source node is active, and created on the destination node. If the source node is down during the migration, the clean-up on the source node happens when the node comes back up.

The following example shows how to migrate all deployments from the source node to the destination node:

```
nfvis# cluster cluster1 migrate-deployment source-node 209.165.201.1 destination-node
209.165.201.2 all-deployments
```

The following example shows how to migrate specific deployments from the source node to the destination node:

```
nfvis# cluster cluster1 migrate-deployment source-node 209.165.201.1 destination-node
209.165.201.2 deployment-list [ centosvirtual machine4 centosvirtual machine5 ]
```

For information about verifying virtual machine migration, see Verify Storage Virtualization.

Virtual Machine Export and Import

Virtual Machines can be exported to any **local/external datastore/NFS** mount path, and can be imported from the **/mnt/gluster/uploads** folder.

Note You cannot export Virtual Machines to the **/mnt/gluster/uploads** folder.

The following example shows how to export a virutal machine:

```
nfvis# vmExportAction exportName asaexport exportPath intdatastore:/uploads/ vmName
ASAvDep.ASAvMgrp
```

The following example shows how to import a virtual machine:

```
nfvis# vmImportAction importPath /mnt/gluster/uploads/backup_router.vmbkp

(or)

nfvis# vmImportAction importPath gluster:/uploads/backup_router.vmbkp
```

Delete the Cluster

Deleting a cluster is similar to cluster creation. The API for cluster deletion has a “no” prefix followed by **cluster-name**. The delete cluster API should be sent to all the nodes in the cluster to delete the cluster.

Delete the virtual machine images and deployments on the cluster before deleting the cluster.

Note After a delete cluster API is sent to a node, sending another create cluster API to the same node does not add the node back to the cluster.

The delete a cluster, use the following command:

```
nfvis(config)# no cluster cluster1
```

Cluster File Operations

Use the following commands to manage the files in the glusterFS storage.

To copy files using the file-copy command:

```
nfvis# system file-copy source /data/intdatastore/uploads/test_file_in_intdatastore.txt
destination /mnt/gluster/uploads
```

To copy files using the scp command:

```
scp into /mnt/gluster/uploads/
nfvis# scp user@172.16.0.1:/nobackup/userdir/test.py gluster:test.py

scp from /mnt/gluster/uploads/
nfvis# scp gluster:test.py user@172.16.0.1:/nobackup/userdir/

scp from outside
nfvis# scp -P 22222 test.txt admin@172.16.0.2:/mnt/gluster/uploads/
```

To view the files present in glusterFS:

```
nfvis# show system file-list disk gluster
SI NO  NAME              PATH                  SIZE  TYPE        DATE MODIFIED
-------------------------------------------------------------------------------------
1      TinyLinux.tar.gz  /mnt/gluster/uploads  17M   VM Package  2022-01-18 16:21:03
```

To delete files from /mnt/gluster/uploads:

```
nfvis# system file-delete file name /mnt/gluster/uploads/test.txt
```

Note The **file-copy** and **file-delete** commands only work for the following folders:

- /data/intdatastore/
- /data/intdatastore/uploads/
- /mnt/extdatastore1/
- /mnt/extdatastore1/uploads/
- /mnt/extdatastore2/
- /mnt/extdatastore2/uploads/
- /mnt/gluster/uploads/ folder

Support Commands for NFVIS Clustering Capability

The following support commands can be used to debug NFVIS clustering:

The following example shows how to check the etcd status:

```
nfvis# support show etcd
Possible completions:
  all-information   Display all etcd information
  cluster-health    Display cluster health
  member-list       Display members of the cluster
  service-status    Display status of etcd services
```

The following example shows how to check the glusterFS status:

```
nfvis# support show gluster
Possible completions:
  all-information   Display all glusterfs information
  mount             Display the glusterfs mount information
  peer-status       Display the glusterfs peer connectivity information
  service-status    Display status of glusterfs services
  volume-info       Display the glusterfs volume information
```

Verify NFVIS Clustering Capability

The following example shows the success scenario of NFVIS clustering capability:

```
nfvis# show cluster
cluster cluster1
node-address 209.165.201.1
cluster-state ok
creation-time 2022-01-18T15:32:23-00:00
details "Gluster peers Healthy"
disk-usage total 10.0
disk-usage available 10.0
                              DEPLOYMENT
ADDRESS           DEPLOYMENT  STATE
----------------------------------------
209.165.201.1
209.165.201.2
209.165.201.3
```

The following example shows the cluster degraded scenario:

```
nfvis# show cluster
cluster cluster1
node-address 209.165.201.1
cluster-state degraded
creation-time 2022-01-18T15:32:23-00:00
details "One Node:
209.165.201.3 in the cluster is down"
disk-usage total 10.0
disk-usage available 10.0
                              DEPLOYMENT
ADDRESS           DEPLOYMENT  STATE
----------------------------------------
209.165.201.1
209.165.201.2
209.165.201.3
```

The following example shows the cluster down scenario:

```
nfvis# show cluster
cluster cluster1
 node-address 209.165.201.1
 cluster-state down
  creation-time 2022-01-18T15:32:23-00:00
 details       "Etcd is inactive"
 disk-usage total 10.0
 disk-usage available 10.0
                         DEPLOYMENT
ADDRESS      DEPLOYMENT  STATE
-------------------------------------
209.165.201.1
209.165.201.2
209.165.201.3
```

The following example shows a successful VM migration:

```
nfvis#  show cluster migration-status
cluster cluster1
migration-status start-time 2022-01-19T14:48:28
migration-status source-node 209.165.201.1
migration-status destination-node 209.165.201.2
migration-status status MIGRATION-SUCCESS
migration-status details ""
DEPLOYMENT STATUS   LAST UPDATE DETAILS
-----------------------------------------------------------------------------------------------
CentosVM   MIGRATED 2022-01-19  22:48:56.897 Migrated VM[CentosVM] to new host[209.165.201.1]
```

```
OTHER56    MIGRATED 2022-01-19  22:48:57.214 Migrated VM[OTHER56] to new host[209.165.201.2]
```



Note In NFVIS 4.8, the **show cluster** command only displays the latest migration status.

NFVIS Clustering Capability Notifications and Syslogs

Table 51: Notifications

Description	Notification	Seen on
Cluster creation in progress	notification eventTime 2022-01-27T12:48:34.440959+00:00 nfvisEvent user_id admin config_change false transaction_id 0 status SUCCESS status_code 0 status_message Cluster creation in progress details NA event_type CLUSTER_CREATION_IN_PROGRESS severity INFO host_name nfvis ! !	All nodes
Cluster creation success	notification eventTime 2022-01-27T12:49:24.068755+00:00 nfvisEvent user_id admin config_change false transaction_id 0 status SUCCESS status_code 0 status_message Cluster creation succeeded details Storage virtualization successful event_type CLUSTER_CREATION_SUCCESS severity INFO host_name nfvis ! !	Leader node

Description	Notification	Seen on
Cluster creation failure	notification eventTime 2022-02-02T19:43:38.281773+00:00 nfvisEvent user_id admin config_change false transaction_id 0 status FAILURE status_code 0 status_message Cluster creation failed details etcd cluster creation failed ETCD cluster health is not healthy event_type CLUSTER_CREATION_FAILURE severity INFO host_name nfvis ! !	All nodes
Cluster deletion in progress	notification eventTime 2022-01-27T04:42:08.654507+00:00 nfvisEvent user_id admin config_change false transaction_id 0 status SUCCESS status_code 0 status_message Cluster deletion in progress details NA event_type CLUSTER_DELETION_IN_PROGRESS severity INFO host_name nfvis ! !	All nodes
Cluster deletion success	notification eventTime 2022-01-27T04:42:12.768702+00:00 nfvisEvent user_id admin config_change false transaction_id 0 status SUCCESS status_code 0 status_message Cluster deletion succeeded details Cluster deletion successful event_type CLUSTER_DELETION_SUCCESS severity INFO host_name nfvis ! !	Node where deletion succeeded

Description	Notification	Seen on
Cluster deletion failure	notification eventTime 2021-12-03T11:35:55.477465+00:00 nfvisEvent user_id admin config_change false transaction_id 0 status FAILURE status_code 0 status_message Cluster deletion failed details ['Internal error'] event_type CLUSTER_DELETION_FAILURE severity INFO host_name nfvis !	Node where deletion failed
VM migration initiated	notification eventTime 2022-01-26T19:49:14.06+00:00 vmlcEvent status SUCCESS status_code 200 status_message Migration request accepted vmlcEvent migration vmlcEvent migration vm name Centos_101 status MIGRATING ! vmlcEvent migration vm name ASAv_101 status MIGRATING ! ! vmlcEvent event type VM_MIGRATE_INIT ! ! !	Destination node for migration
VM migration completed	notification eventTime 2022-01-26T19:49:46.139+00:00 vmlcEvent status SUCCESS status_code 200 status_message Migration processing complete vmlcEvent migration vmlcEvent migration vm name Centos_101 status MIGRATION-SUCCESS ! vmlcEvent migration vm name ASAv_101 status MIGRATION-SUCCESS ! ! vmlcEvent event type VM_MIGRATE_COMPLETE ! ! !	Destination node for migration

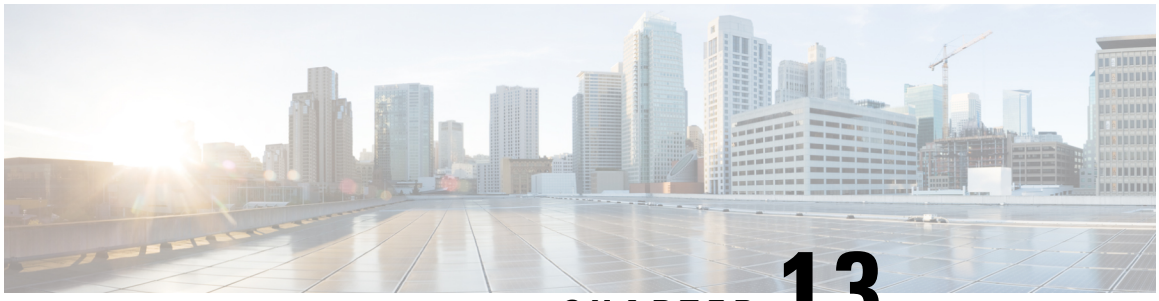
Description	Notification	Seen on

VM migration failed at source	notification
	 eventTime 2022-01-26T00:53:19.737912+00:00
	 nfvisEvent
	  user_id admin
	  config_change false
	  transaction_id 0
	  status FAILURE
	  status_code 0
	  status_message Please stop Deployments ['ASAv_209', 'OTHER18', 'c8kv_209'] before migration
	  details NA
	  event_type MIGRATION_FAILURE
	  severity INFO
	  host_name 172-25-221-9
	 !
	!
		At source if validation fails at source

Syslogs

Description	Syslog	Sent from
Cluster creation in progress	Jan 27 04:48:34 nfvis %SYS-6-CLUSTER_CREATION_IN_PROGRESS: Cluster creation in progress	All nodes
Cluster creation success	Jan 27 04:49:24 nfvis %SYS-6-CLUSTER_CREATION_SUCCESS: Cluster creation succeeded Storage virtualization successful	Leader node
Cluster creation failed	Jan 26 21:12:34 nfvis %SYS-3-CLUSTER_CREATION_FAILURE: Cluster creation failed etcd cluster creation failed ETCD cluster health is not healthy	All nodes
Cluster deletion in progress	Jan 27 04:45:13 nfvis %SYS-6-CLUSTER_DELETION_IN_PROGRESS: Cluster deletion in progress	All nodes
Cluster deletion success	Jan 27 04:45:16 nfvis %SYS-6-CLUSTER_DELETION_SUCCESS: Cluster deletion succeeded Cluster deletion successful	Node where the deletion succeeded
Cluster delete failed	Jan 26 11:36:55 nfvis %SYS-3-CLUSTER_DELETION_FAILURE: Cluster deletion failed ['Internal error']	Node where the deletion failed
VM migration initiated	Jan 26 11:49:14 172-25-221-101 %SYS-6-VM_MIGRATE_INIT: VM migration initiation successful: Centos_101 MIGRATING ASAv_101 MIGRATING	Destination node for migration
VM migration completed	Jan 26 11:49:47 172-25-221-101 %SYS-6-VM_MIGRATE_COMPLETE: VM migration successful: Centos_101 MIGRATION-SUCCESS ASAv_101 MIGRATION-SUCCESS	Destination node for migration

Description	Syslog	Sent from
VM migration failed at source	`Jan 25 16:53:19 172-25-221-9 %SYS-3-MIGRATION_FAILURE: Please stop Deployments ['ASAv_209', 'OTHER18', 'c8kv_209'] before migration`	At source if validation fails at source



CHAPTER 13

Platform Specific Configurations

Table 52: Feature History

Feature Name	Release Information	Description
Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE	Cisco NFVIS Release 4.16.1	The Datapath Over LTE Cellular Interface feature on the Cisco Catalyst 8200 UCPE facilitates traffic routing through an LTE cellular interface. This feature allows both control connection and virtual machine data traffic to be routed via the cellular interface, providing redundancy and an alternative path for wired connections.
C8300-UCPE Virtual RAID and Supported NIMs Specifications	Cisco NFVIS Release 4.18.1	The C8300-UCPE platform has new functionalities: • It introduces C-NIM-8T and C-NIM-4X network interface modules (NIMs). • It supports a software Redundant Array of Independent Disks (RAID) controller, which allows you to create a RAID array using two identical E-SSD-U2N disks (U.2 NVMe drives), configured as either RAID1 or RAID0.

- ENCS Switch Configuration, on page 274
- Configure Additional WAN Transport Connectivity On ENCS Switch Ports, on page 284
- Configuring vBranch High Availability, on page 289
- CIMC Secure Overlay Support, on page 301
- CIMC TACACS Support, on page 302

- LTE PIM Module on Cisco Catalyst 8200 UCPE, on page 303
- Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE, on page 322
- External Storage for Cisco ENCS 5400 and Cisco Cloud Services Platforms, on page 325
- Support for 40G Dual Port and Quad-Split NICs on Cisco Cloud Services Platforms, on page 327
- C8300-UCPE Virtual RAID and Supported NIMs Specifications, on page 330

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

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

POR T	IN-UCAS T	OUT-UCAS T	IN-MCAS T	OUT-MCAS T	IN-BCAS T	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 53: Switch Settings Interface

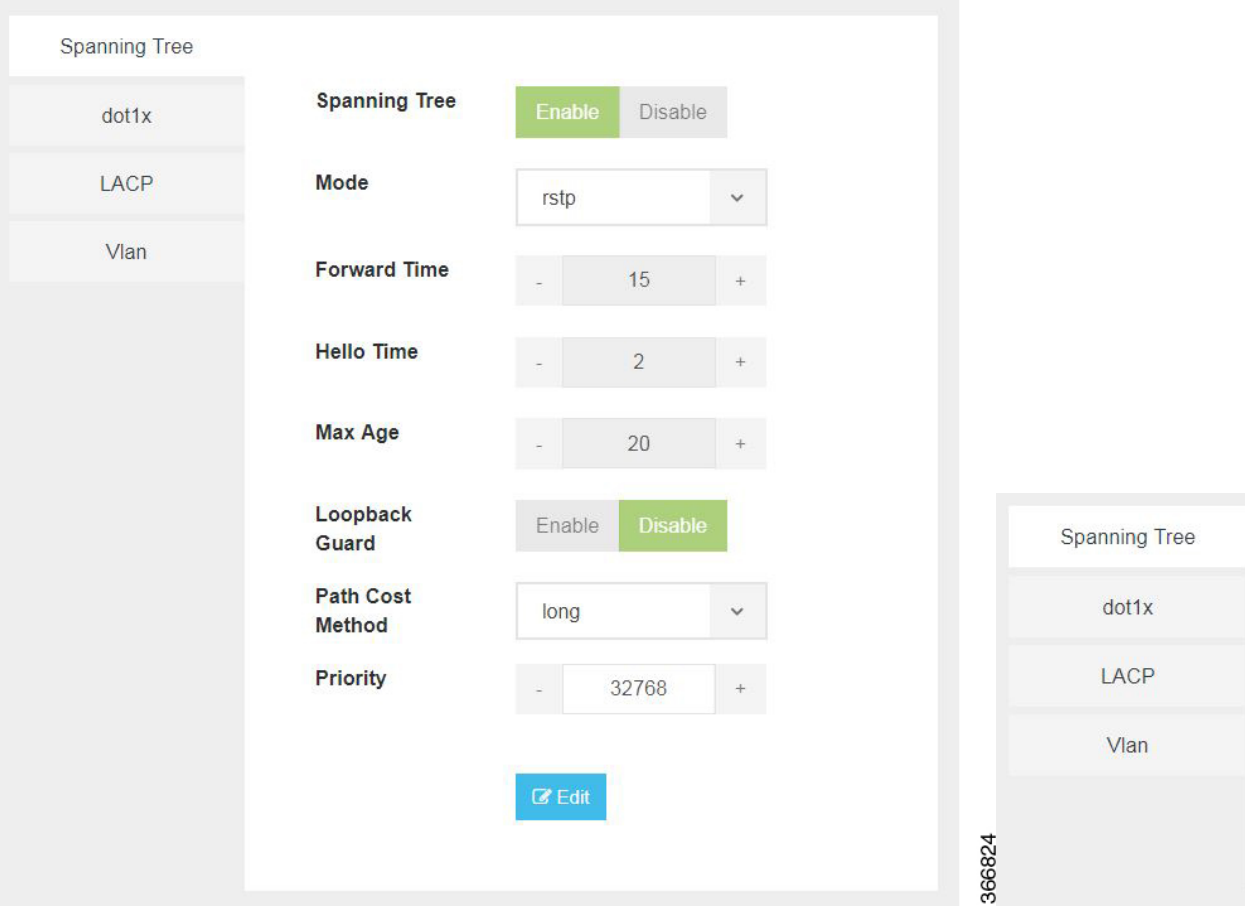
Parameter	Description	Values
SwitchPort	Specifies the switch interface name.	
Description	Specifies the description of the interface.	
Status	Specifies the status of the interface.	up or down
MAC Address	Specifies the MAC address of the interface.	
PortType	Specifies the mode of the port interface.	Supported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

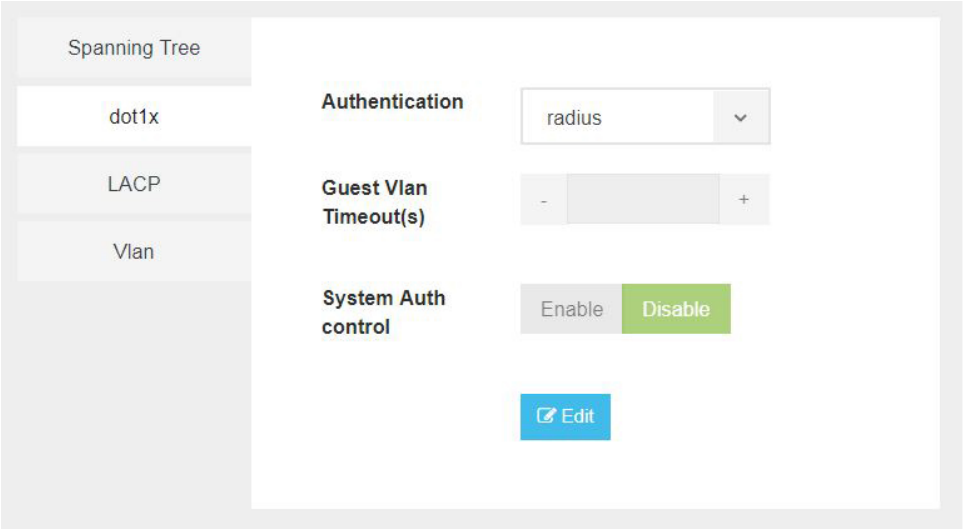
Table 54: Spanning Tree Parameters

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:

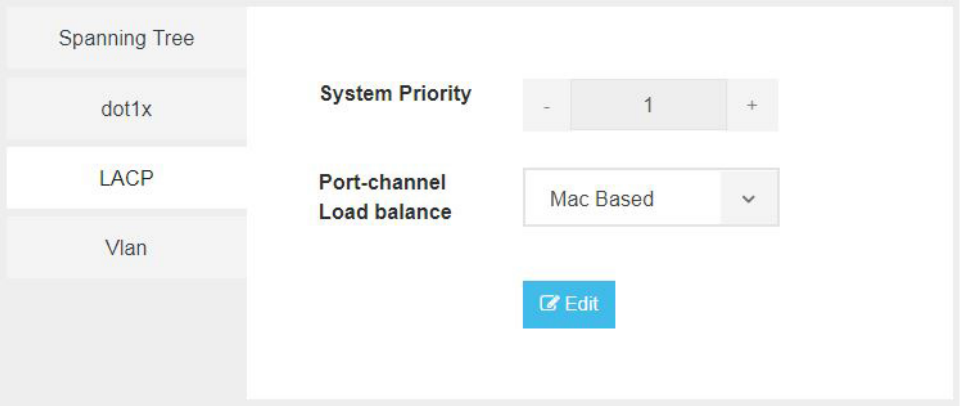
Table 55: Dot1x Parameters

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:

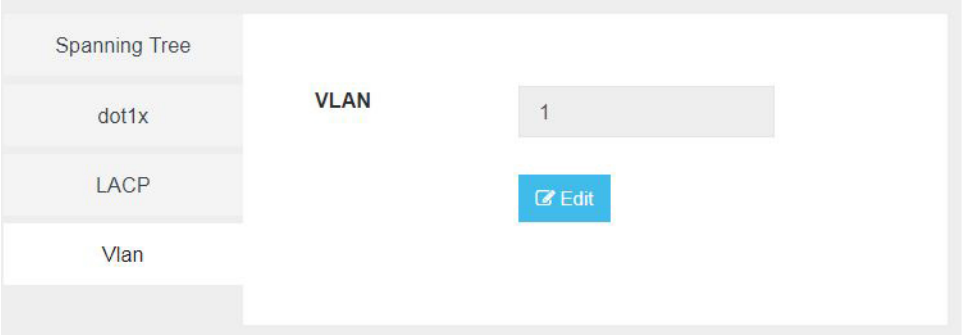
Table 56: LACP Parameters

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.

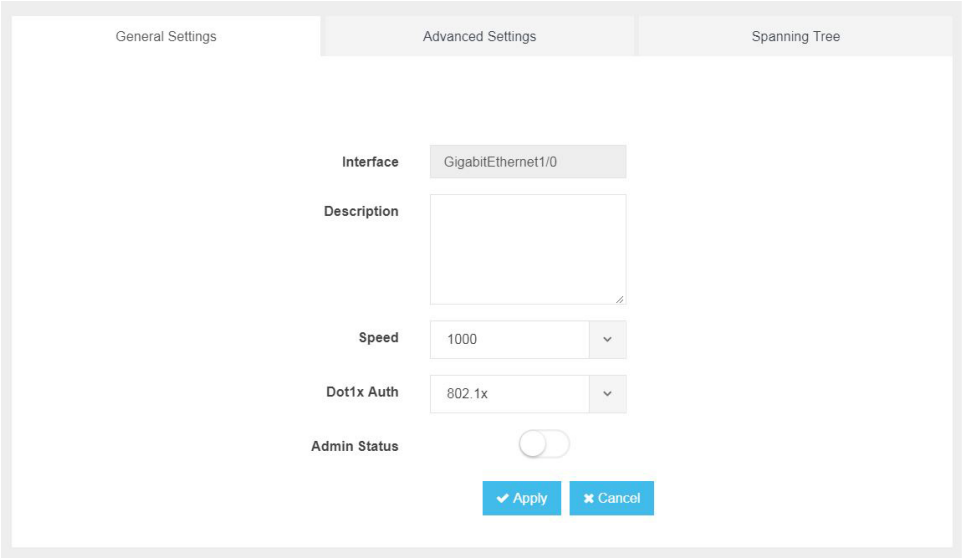
You can configure VLANs in the range **<1-2349>|<2450-4093>** for a specified switch port.



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

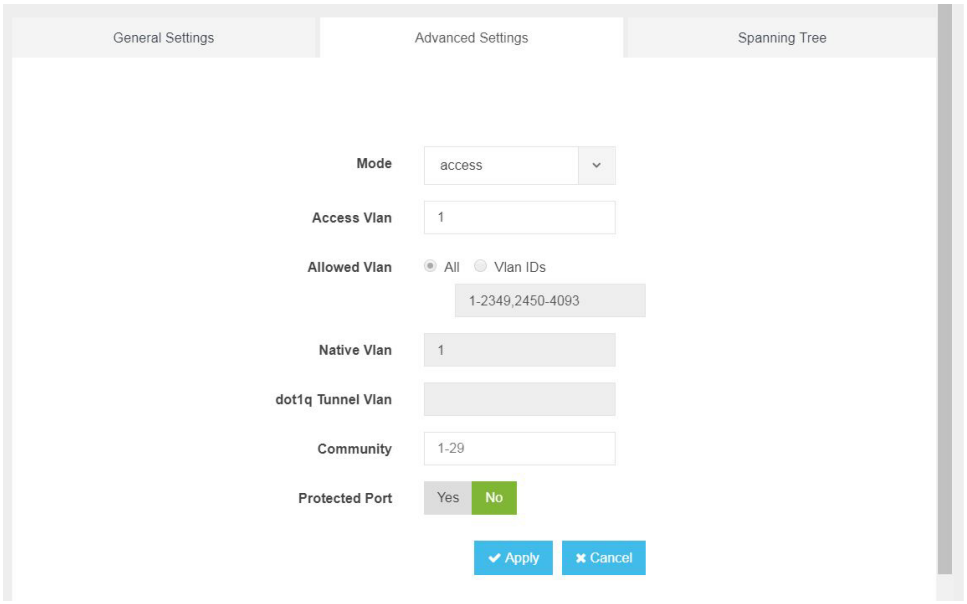


You can configure general settings using the following parameters for each switch interface:

- 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

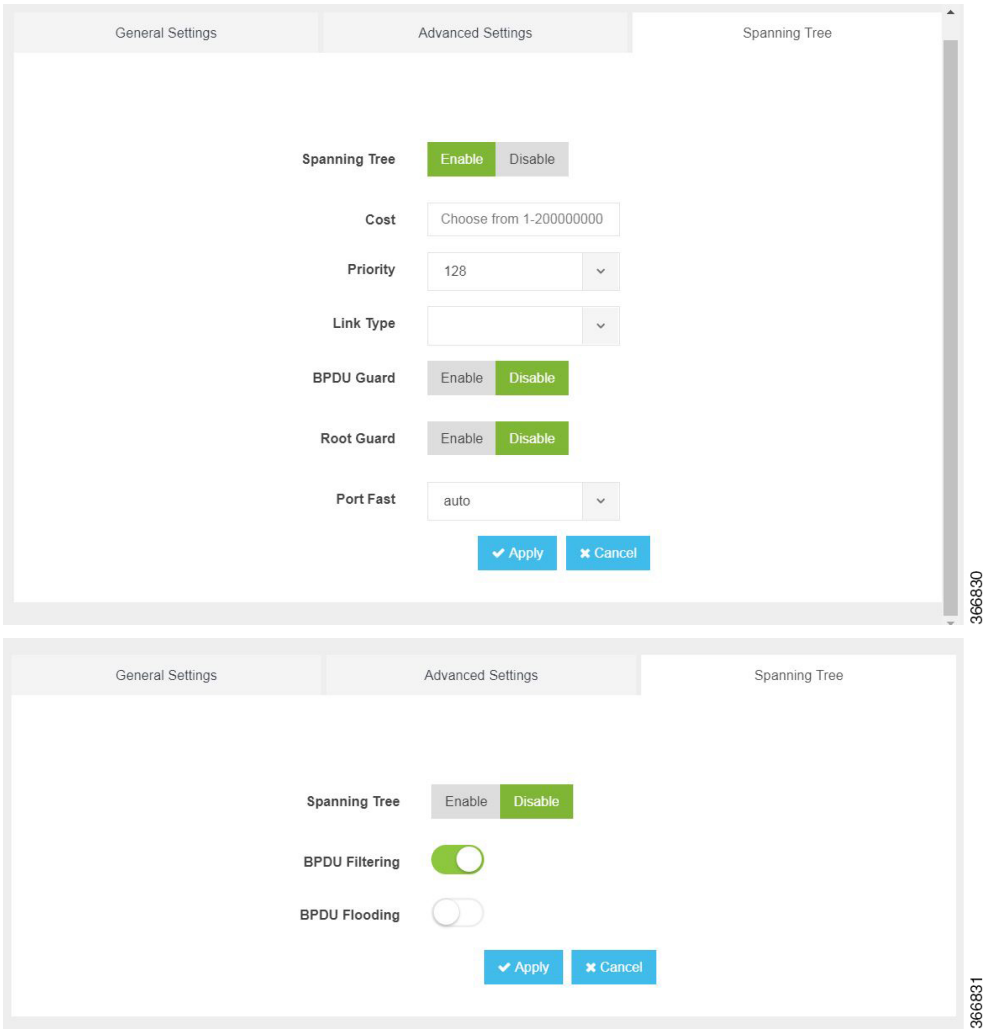


You can make the advanced settings using the following parameters for each switch interface:

- 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 279.

Configuring Spanning Tree per Interface



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

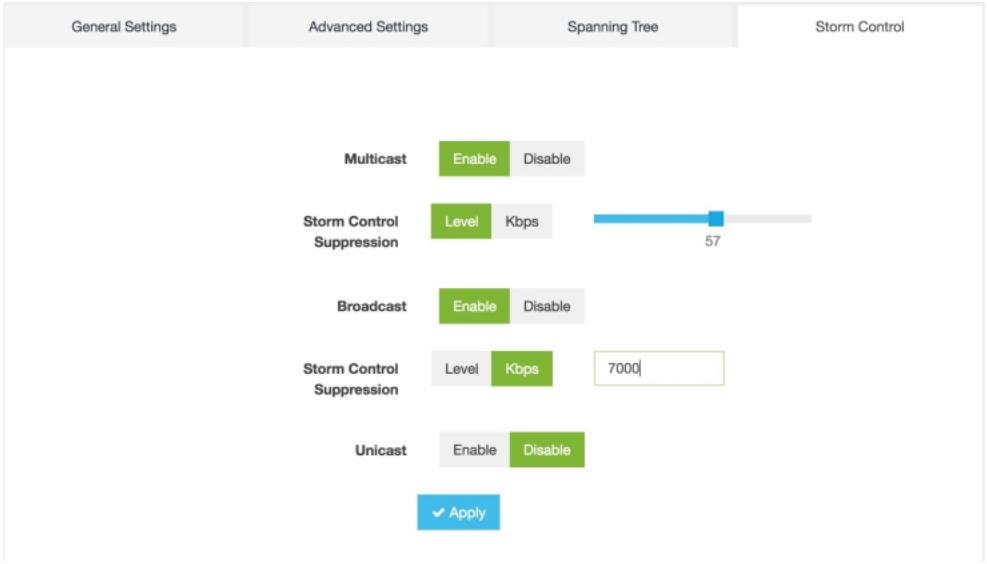
- Spanning Tree—Enable or Disable
- Cost—Specifies the cost. Range: 1 to 200000000
- Priority—Specifies the port priority. Range: 0 to 240, default value is 128
- Link Type—point-to-point or shared
- BPDU Guard—Enable or Disable
- Root Guard—Enable or Disable

- 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

Configure Storm Control

Storm control is used to monitor incoming traffic levels and limit excessive flow of packets on any user facing switch port that could cause a traffic storm. Traffic storms can lead to device instability and unintended behavior.

You can configure storm control from Cisco NFVIS Portal, from **Storm Control** tab.



Storm control can be configured for specific type of traffic - unicast or multicast or broadcast. The suppression range can be in terms of a percentage level (1-100) or Kbps value (1-1000000).

Configure Additional WAN Transport Connectivity On ENCS Switch Ports

Table 57: Feature History

Feature Name	Release Information	Description
Configure Additional WAN Transport Connectivity On ENCS Switch Ports	NFVIS Release 4.10	This feature enables configuring additional Transport Locator (TLOC) an attachment point. The attachment point is where a Cisco WAN Edge device connects to a WAN transport using ENCS LAN ports as WAN ports. This is in addition to the currently available WAN connectivity options that is restricted to just two WAN ports.

Starting from NFVIS Release 4.10, map LAN SR-IOV (Single Root Input or Output Virtualization) with Cisco ISRv or Cisco Catalyst 8000V to configure additonal WAN transport connectivity on ENCS switch ports. The following are the changes to ENCS LAN and WAN ports connectivity:

- The WAN circuit TLOC terminates on layer 3 ports (GE0-0 or GE0-1).
- The WAN circuit TLOC physically connects to any switch port (GE1/0 - GE1/7).
- The LAN traffic flows through the switchports (GE1/0 - GE1/7).

Note Cisco Security Group Tag (SGT) is added to the LAN interfaces which are propagated to all the TLOCs that are handled by VNFs such as Cisco ISRv or Cisco Catalyst 8000V. For more information see, Cisco TrustSec Integration.

Prerequisites to Configure Additional WAN Transport Connectivity On ENCS Switch Ports

Setup Cisco ISRv or Cisco Catalyst 8000V devices using Cisco NFVIS.

Configure Additional WAN Transport Connectivity Using MPLS

This section provides an example configuration to configure additional WAN ports on ENCS switch ports with ISRv and Catalyst 8000V using MPLS:

1. Enter the configuration mode:

   ```
   config-transaction
   ```

2. Enter the sdwan mode:

```
sdwan
```

3. Configure the interface into a tunnel interface:

```
tunnel interface
```

4. Configure an interface type and enter the interface configuration mode:

```
interface GigabitEthernet2
```

5. Configure an encapsulation for a tunnel interface:

```
encapsulation ipsec
```

6. Assign a color to a WAN transport tunnel:

```
color mpls
```

7. Configure the services that are allowed on a tunnel interface:

```
allow-service all
no allow-service bgp
allow-service dhcp
allow-service dns
allow-service icmp
no allow-service sshd
no allow-service netconf
allow-service ntp
no allow-service ospf
no allow-service stun
allow-service https
no allow-service snmp
no allow-service bfd
```

Here's the complete configuration example for configuring additional WAN ports on ENCS switch ports with ISRv and Catalyst 8000V using MPLS:

```
sdwan
interface GigabitEthernet2
tunnel-interface
encapsulation ipsec
color mpls
max-control-connections 0
allow-service all
no allow-service bgp
allow-service dhcp
allow-service dns
allow-service icmp
no allow-service sshd
no allow-service netconf
allow-service ntp
no allow-service ospf
no allow-service stun
allow-service https
no allow-service snmp
no allow-service bfd
!
```

Configure Additional WAN Transport Connectivity Using Biz Internet

This section provides an example configuration to configure additional WAN ports on ENCS switch ports with ISRv and Catalyst 8000V using biz internet:

1. Enter the configuration mode:

   ```
   config-transaction
   ```

2. Enter the sdwan mode:

   ```
   sdwan
   ```

3. Configure the interface into a tunnel interface:

   ```
   tunnel interface
   ```

4. Configure an interface type and enter the interface configuration mode:

   ```
   interface GigabitEthernet5.101
   ```

5. Configure an encapsulation for a tunnel interface:

   ```
   encapsulation ipsec
   ```

6. Restrict the biz internet colors:

   ```
   color biz internet restrict
   ```

7. Configure the services that are allowed on a tunnel interface:

   ```
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
   no allow-service ntp
   no allow-service ospf
   no allow-service stun
   allow-service https
   no allow-service snmp
   no allow-service bfd
   ```

Here's the complete configuration example for configuring additional WAN ports on ENCS switch ports with ISRv and Catalyst 8000V using biz internet:

```
sdwan
interface GigabitEthernet5.101
tunnel-interface
encapsulation ipsec
color biz-internet restrict
no allow-service bgp
allow-service dhcp
allow-service dns
allow-service icmp
no allow-service sshd
no allow-service netconf
no allow-service ntp
no allow-service ospf
no allow-service stun
allow-service https
no allow-service snmp
no allow-service bfd
!
```

Configure Additional WAN Transport Connectivity Using Public Internet

This section provides an example configuration to configure additional WAN ports on ENCS switch ports with ISRv and using public internet:

1. Enter the configuration mode:

   ```
   config-transaction
   ```

2. Enter the sdwan mode:

   ```
   sdwan
   ```

3. Configure the interface into a tunnel interface:

   ```
   tunnel interface
   ```

4. Configure an interface type and enter the interface configuration mode:

   ```
   interface GigabitEthernet6
   ```

5. Configure an encapsulation for a tunnel interface:

   ```
   encapsulation ipsec
   ```

6. Restrict the public internet colors:

   ```
   color public internet restrict
   ```

7. Configure the services that are allowed on a tunnel interface:

   ```
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
   allow-service ntp
   no allow-service ospf
   no allow-service stun
   allow-service https
   no allow-service snmp
   no allow-service bfd
   ```

Here's the complete configuration example for configuring additional WAN ports on ENCS switch ports with ISRv and Catalyst 8000V using public internet:

```
sdwan
interface GigabitEthernet6
tunnel-interface
encapsulation ipsec
color public-internet restrict
no allow-service bgp
allow-service dhcp
allow-service dns
allow-service icmp
no allow-service sshd
no allow-service netconf
allow-service ntp
no allow-service ospf
no allow-service stun
allow-service https
no allow-service snmp
```

```
no allow-service bfd
!
```

Configuration Example of Additional WAN Transport Connectivity to Cisco ISRv or Catalyst 8000V

The following example shows the interfaces mapped to Cisco ISRv or Cisco Catalyst 8000V VM:

```
Name: S200-CE1
Deployment Name : S200-CE1
VM Group Name : S200-CE1
State: ALIVE
Internal State: VM_ALIVE_STATE
Bootup Time: 600
Image: isrv-universalk9.17.03.03.tar.gz
Flavor: ISRv-medium

VCPU# Memory(MB) Disk(MB)
----------------------------
4 4096 8192

Low Latency: true
VCPU CPU CORE SOCKET
-----------------------
0 2 2 0
1 3 3 0
2 4 4 0
3 5 5 0

NICID VNIC NETWORK IP MAC-ADDRESS MODEL PORT-FORWARD
-------------------------------------------------------------------------------------------
0 vnic0 int-mgmt-net 10.20.0.2 52:54:00:cf:41:cc virtio  <----- Internal Monitoring
1 vnic1 LAN-SRIOV-20 - 52:54:00:90:3b:8c None <---- connects to GE1/0 for TLOC MPLS
2 vnic2 LAN-SRIOV-21 - 52:54:00:8a:c1:a1 None <---- host on LAN (service VPN/VRF)
3 vnic3 LAN-SRIOV-22 - 52:54:00:9c:17:db None <---- host on LAN (service VPN/VRF)
4 vnic4 LAN-SRIOV-23 - 52:54:00:f6:12:eb None <---- connects to GE1/6 for TLOC Biz-Internet
5 vnic5 GE0-0-SRIOV-1 - 52:54:00:e8:19:74 None <---- connects to GE0-0 for TLOC
Public-Internet
6 vnic6 GE0-1-SRIOV-1 - 52:54:00:53:d4:d7 None <------ unused
```

Verify Mapping of LAN SR-IOV with Cisco ISRv or Cisco Catalyst 8000V Using Cisco ISRv or Cisco Catalyst 8000V CLI

The following is a sample output from the **show sdwan control connections** command:

```
router# show sdwan control connections
PEER PEER CONTROLLER
PEER PEER PEER SITE DOMAIN PEER PRIV PEER PUB GROUP
TYPE PROT SYSTEM IP ID ID PRIVATE IP PORT PUBLIC IP PORT LOCAL COLOR PROXY STATE UPTIME ID

------------------------------------------------------------------------------------------------------------------------
vsmart dtls 10.0.0.201 10000 1 192.10.10.200 12446 192.10.10.200 12446 public-internet No
up 19:00:27:09 0
vbond dtls 0.0.0.0 0 0 10.10.0.3 12346 10.10.0.3 12346 biz-internet - connect 0
vbond dtls 0.0.0.0 0 0 10.10.0.3 12346 10.10.0.3 12346 public-internet - up 19:00:27:09 0
vmanage dtls 10.0.0.101 10000 0 192.10.10.100 12446 192.10.10.100 12446 public-internet No
 up 19:00:27:09 0
```

Configuring vBranch High Availability

High availability design provides redundancy for WAN, LAN, ENCS device, vRouter, vFirewall VNF level redundancy.

A branch site can have two routers for redundancy. If vEdge-cloud router is chosen, Each of the vedge-cloud router maintains:

- A secure control plane connection, via a DTLS connection, with each vSmart controller in its domain
- A secure data plane connection with the other vEdge routers at the site

Because both vEdge routers receive the same routing information from the vSmart controllers, each one is able to continue to route traffic if one should fail, even if they are connected to different transport providers.

Two firewalls are placed in a group and their configuration is synchronized to prevent a single point of failure on your network. A heartbeat connection between the firewall peers ensures seamless failover in the event that a peer goes down. Setting up two firewalls in an HA pair provides redundancy and allows you to ensure business continuity.

Prerequisites for vBranch HA

The WAN links are active on both Cisco ENCS1 and Cisco ENCS2. Each of the ENCS WAN link is connected to the WAN network (most cases with two SPs), with two ENCSs in an active-active mode.

The LAN facing links of both Cisco ENCS devices are connected to an external switch (as an uplink), and all the devices on the LAN segment are also connected to the external switch. There should be no LAN device connecting directly to the Cisco ENCS internal switch.

Two vRouters and the Two vFirewalls have full mesh L3 connectivity.

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

SD-Branch HA Design and Topology

In HA design, there are two sets of VLANs. Traffic path is between the VNFs and traffic from or towards LAN.

To protect against cable connection issue and box failure, there is back-to-back cable between ENCS and connection from each ENCS to the external switch.

When using Cisco ENCS and Cisco switches, common expectation is to use PVST+, detect loops and switch specific ports to BLOCKING mode. ENCS switch does not support PVST (Per VLAN spanning tree). By Default, RSTP could end up blocking ENCS port back-to-back connection, this will result in blocking traffic path between the VNFs.

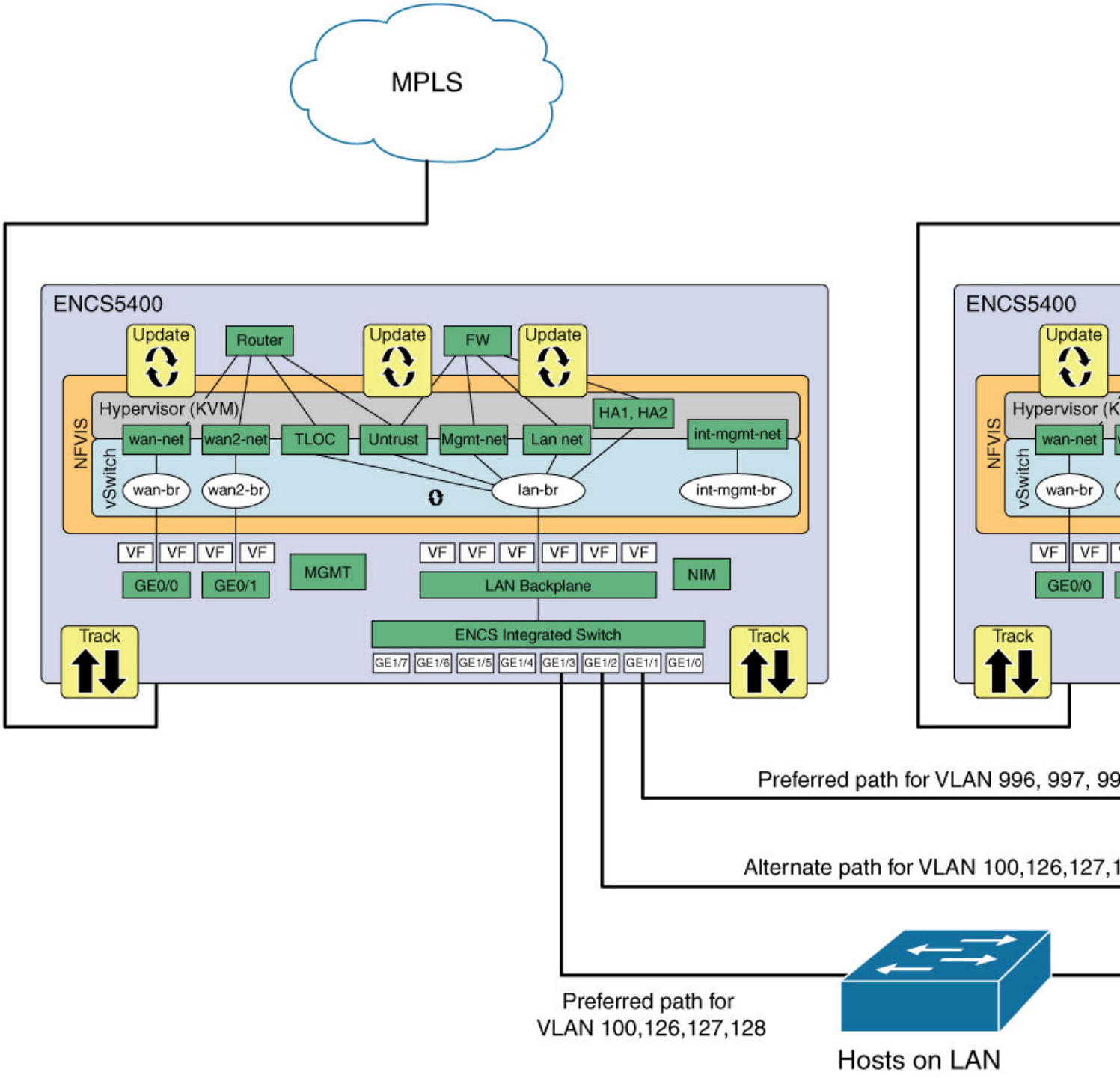
The recommended solution is to use MSTP in ENCS and the external switches. The following topology and configuration provides a step-by-step procedure with reasoning for specific configuration use. There are two instances of MSTP created. One for handling traffic path between VNFs and the second for handling traffic from or towards LAN.



Note In cases where external switch cannot be configured for MSTP, RSTP is used and the two links back-2-back between ENCS is not in port-channel.

- One of the links carries traffic between VNFs by configuring disable spanning tree. The second back-to-back link between ENCS processes RSTP and forward or block for the traffic from or towards LAN.
- From each of the ENCS, a third physical link connects to the external switch. This also forwards or blocks the traffic from or towards LAN depending on the RSTP decisions.

Physical Device Connections



VM and Service Chain Network Connection

Figure 7: ENCS-Left

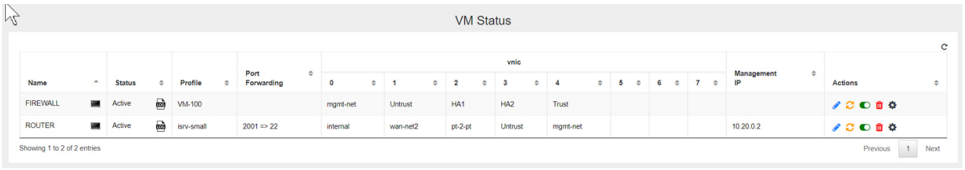
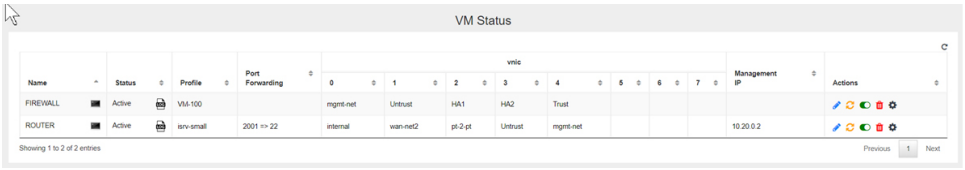


Figure 8: ENCS-Right



Note In the absence of firewall in the design, the router is directly connected to the LAN side. Pt-to-Pt network extends the TLOC connection across the ENCS devices and VRRP is enabled in the router LAN facing connection.

Isolating LAN and Transit Link Traffic for vBranch HA

Traffic from or towards LAN and traffic between the VNFs are isolated by configuring different VLANs for each traffic since both links are connected to the same ENCS internal switch. If you do not isolate the traffic, both LAN traffic and transit link will flow through the same internal switch on the Cisco ENCS.

Enable Port Tracking and Virtual NIC Update

The configured VNICs tracks the state of the ports based on the PNICs notifications. To verify the state of the port, use **show interface** or **ethtool** commands. You can also use commands specific to the VM, that displays the interface link state.

To configure track state on GE0-0 & GE0-1:

```
configure terminal
pnic GE0-0 track-state ROUTER 1
end
```

```
ENCS-Left# support show ifconfig GE0-0

GE0-0: flags=4611<UP,BROADCAST,ALLMULTI,MULTICAST>  mtu 9216
        ether 70:db:98:c3:df:28  txqueuelen 1000  (Ethernet)
```

To configure track state on switch port:

```
configure terminal
switch interface gigabitEthernet 1/3 track-state FIREWALL 4
end
```

```
ENCS-Left# show vm_lifecycle deployments FIREWALL


Name: FIREWALL
Deployment Name : FIREWALL
VM Group Name : FIREWALL
State: ALIVE
Internal State: VM_INERT_STATE
Bootup Time: -1
Image: Palo-Alto-8.1.3.tar.gz
Flavor: VM-100


VCPU#  Memory(MB)  Disk(MB)
----------------------------
2      7168        61440


Low Latency: true
VCPU  CPU  CORE  SOCKET
----------------------
0     3      3     0
1     2      2     0


NICID  VNIC   NETWORK  IP   MAC-ADDRESS        MODEL    PORT-FORWARD
-------------------------------------------------------------------------------
0      vnic6  mgmt-net -    52:54:00:2b:72:d2  virtio
1      vnic7  Untrust  -    52:54:00:eb:a3:e7  virtio
2      vnic8  HA1      -    52:54:00:f4:de:e5  virtio
3      vnic9  HA2      -    52:54:00:12:f8:21  virtio
4      vnic10  Trust    -    52:54:00:7a:6b:e9  virtio



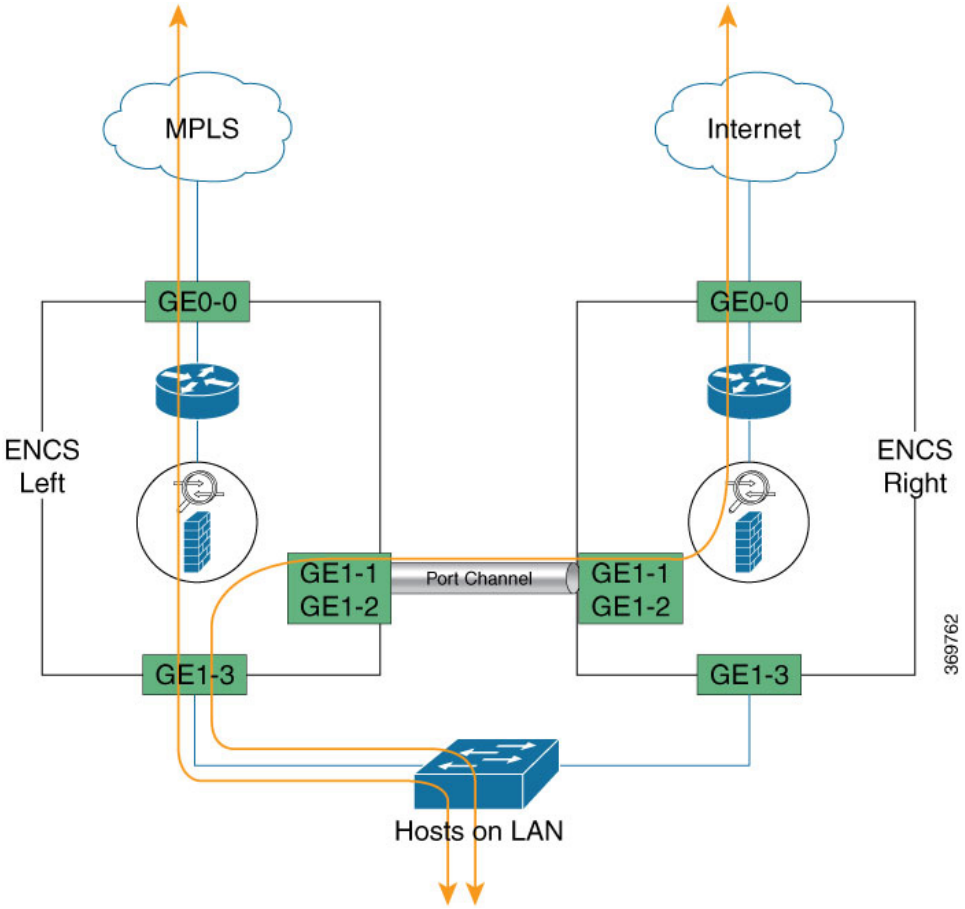
ENCS-Left# support show ifconfig vnic10

vnic10: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9216
        inet6 fe80::fc54:ff:fe7a:6be9  prefixlen 64  scopeid 0x20<link>
        ether fe:54:00:7a:6b:e9  txqueuelen 4000  (Ethernet)
```

Packet Flow for SD-Branch HA

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

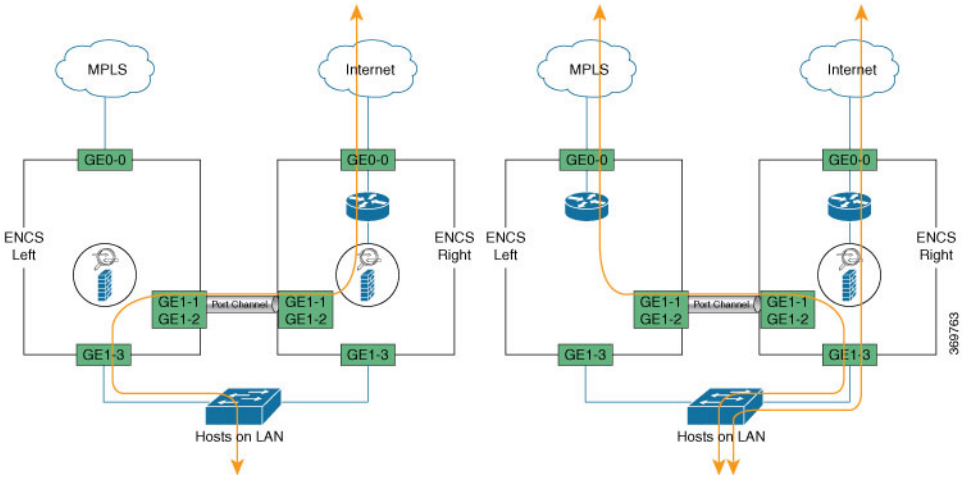
Non-Failure Case



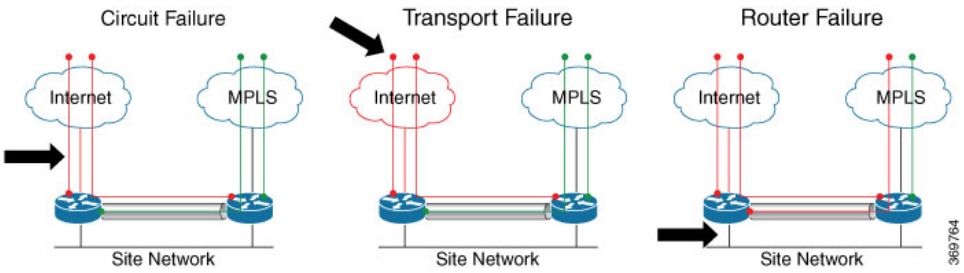
In the non-failure case, both ENCS devices are Active, up and running

- LAN to WAN through the ENCS1 Firewall and ENCS1 Router
- LAN to WAN through the ENCS1 Firewall and ENCS2 Router
- WAN to LAN through ENCS1 Router and ENCS1 Firewall
- WAN to LAN through ENCS2 Router and ENCS1 Firewall

Failure Case



Following are failures that a router must be designed and configured to adapt



The conditions that trigger a firewall failover are:

- One or more of the monitored interfaces fail. (Link Monitoring)
- One or more of the destinations specified on the firewall cannot be reached. (Path Monitoring)
- The firewall does not respond to heartbeat polls. (Heartbeat Polling and Hello messages)

Configuration Examples and Usage Description

ENCS-Left and ENCS-Right with Same Config	Description or Reasons for configuration
networks network wan-net bridge wan-br ! networks network HA1 vlan [126] trunk false bridge lan-br ! networks network HA2 vlan [127] trunk false bridge lan-br ! networks network Trust vlan [128] bridge lan-br ! networks network Untrust vlan [998] bridge lan-br ! networks network mgmt-net vlan [100] trunk false bridge lan-br ! networks network pt-2-pt vlan [996 997] bridge lan-br	In a HA design involving a router or Firewall, there are 3 to 6 paths required. ENCS platform has 2 WAN facing ports and 8 LAN facing ports. • WAN facing ports are reserved for connection to WAN circuits. • LAN facing ports are the only set of available ports for creating the 3 to 6 path required. Between VNFs and LAN, OVS or SR-IOV VFs and physical switch ports are the two Layer2 entities to traverse.
! vlan 1 ! vlan 100 ! vlan 126 ! vlan 127 ! vlan 128 ! vlan 996 ! vlan 997 ! vlan 998 ! spanning-tree enable spanning-tree mode mst spanning-tree mst 2 priority 61440 spanning-tree mst configuration name mst_LAN instance 1 vlan 996-998 instance 2 vlan 100,126-128 !	VLAN must be explicitly created before they are used in the interfaces. Enable MSTP. For MST group 2 carrying “Traffic towards/from LAN”, force the External Switch to become the ROOT using the “mst <group> priority <value>” CLI. The Higher the value, lower the chance of becoming spanning-tree ROOT. “priority” configuration is NOT required for the MST group 1 carrying “Traffic between VNFs”. There is NO loop possibility for MST group 1 VLANs.

ENCS-Left and ENCS-Right with Same Config	Description or Reasons for configuration

nfvis# show running-config switch
switch
interface gigabitEthernet1/1
  no shutdown
  channel-group 1 mode auto
 !
 interface gigabitEthernet1/2
  no shutdown
  channel-group 1 mode auto
!
switch
 interface port-channel1
  negotiation auto
  no shutdown
  spanning-tree mst 1 cost 200000000
  spanning-tree mst 2 cost 200000000
  switchport mode trunk
  switchport trunk native vlan 1
  switchport trunk allowed vlan
100,126-128,996-998
 !
	For the back-to-back ENCS connection, link redundancy is achieved using port-channel configuration. Interfaces that are belong to a port-channel group use configuration from “interface port-channel x”
	
	Goal is to prefer the direct links from ENCS to the External Switch for “Traffic towards/from LAN”. In ENCS back-to-back connection, Spanning tree cost is HIGH for MST group carrying “Traffic towards/from LAN”. This config will block one of the ENCS back-to-back interfaces for breaking the loop for MST group carrying “Traffic towards/from LAN”.

Status of MST instances.

For MST instance 1, “Traffic between the VNFs”, back-to-back portchannel link is root and forwading state.

For MST instance 2, “Traffic from/towards the LAN”, links connected to External Switch are in forwarding state, path via back-to-back portchannel link is “Blocking state”. If one of the Links fail between ENCS and External switch, portchannel path for MST instance 2 will be unblocked.

```
ENCS-Left# show switch vlan detailed

VLAN               TAGGED
ID    VLAN  NAME   PORTS        UNTAGGED PORTS
       CREATED BY
-------------------------------------------------------------------------------
1     1     1      None
gi0,gi4-6,te2,po2-4   DefaultVoiceVLAN
100   100   100    gi3,te2,po1  gi7
         Manual
126   126   126    gi3,te2,po1  None
         Manual
127   127   127    gi3,te2,po1  None
         Manual
128   128   128    gi3,te2,po1  None
         Manual
996   996   996    te2,po1      None
         Manual
997   997   997    te2,po1      None
         Manual
998   998   998    te2,po1      None
         Manual

ENCS-Left# show switch spanning-tree mstp
summary

spanning-tree mstp summary ist-info summary
admin-status enabled
spanning-tree mstp summary ist-info summary
Operation-mode MSTP
spanning-tree mstp summary ist-info summary
Port-Cost-Method long
spanning-tree mstp summary ist-info summary
Loopback-guard disabled
spanning-tree mstp summary ist-info root
Priority 32768
spanning-tree mstp summary ist-info root
Address 70:db:98:c3:df:14
spanning-tree mstp summary ist-info root Cost
 0
spanning-tree mstp summary ist-info root Port
 LAG1
spanning-tree mstp summary ist-info root
Hello-Time 2
spanning-tree mstp summary ist-info root
Max-Age 20
spanning-tree mstp summary ist-info root
Forward-Delay 15
spanning-tree mstp summary ist-info bridge
Priority 32768
spanning-tree mstp summary ist-info bridge
Address 70:db:98:c3:df:a0
spanning-tree mstp summary ist-info bridge
Hello-Time 2
spanning-tree mstp summary ist-info bridge
Max-Age 20
spanning-tree mstp summary ist-info bridge
Forward-Delay 15
spanning-tree mstp summary ist-info
…..
…..

INSTANCE  PRIORITY  DSG ROOT ADDRESS   BRIDGE
```

```
ENCS-Right# show switch vlan detail

VLAN               TAGGED
ID    VLAN  NAME   PORTS        UNTAGGED PORTS
       CREATED BY
-------------------------------------------------------------------------------
1     1     1      None
gi0,gi4-6,te2,po2-4   DefaultVoiceVLAN
100   100   100    gi3,te2,po1  gi7
         Manual
126   126   126    gi3,te2,po1  None
         Manual
127   127   127    gi3,te2,po1  None
         Manual
128   128   128    gi3,te2,po1  None
         Manual
996   996   996    te2,po1      None
         Manual
997   997   997    te2,po1      None
         Manual
998   998   998    te2,po1      None
         Manual

ENCS-Right# show switch spanning-tree mstp
summary

spanning-tree mstp summary ist-info summary
admin-status enabled
spanning-tree mstp summary ist-info summary
Operation-mode MSTP
spanning-tree mstp summary ist-info summary
Port-Cost-Method long
spanning-tree mstp summary ist-info summary
Loopback-guard disabled
spanning-tree mstp summary ist-info root
Priority 32768
spanning-tree mstp summary ist-info root
Address 70:db:98:c3:df:14
spanning-tree mstp summary ist-info root Cost
 0
spanning-tree mstp summary ist-info root Port
 0
spanning-tree mstp summary ist-info root
Hello-Time 2
spanning-tree mstp summary ist-info root
Max-Age 20
spanning-tree mstp summary ist-info root
Forward-Delay 15
spanning-tree mstp summary ist-info bridge
Priority 32768
spanning-tree mstp summary ist-info bridge
Address 70:db:98:c3:df:14
spanning-tree mstp summary ist-info bridge
Hello-Time 2
spanning-tree mstp summary ist-info bridge
Max-Age 20
spanning-tree mstp summary ist-info bridge
Forward-Delay 15
spanning-tree mstp summary ist-info
……
……

INSTANCE  PRIORITY  DSG ROOT ADDRESS   BRIDGE
```

```
 ADDRESS
-------------------------------------------------------------------
1          32768     70:db:98:c3:df:14
70:db:98:c3:df:a0
2          61440     f0:b2:e5:56:e4:80
70:db:98:c3:df:a0

INST                    PRIO.
ID    PORT    STATE     NBR     COST      STS
      ROLE
-------------------------------------------------------------------
1     gi1/0  enabled  128.1  2000000  disabled
    disabled
1     gi1/3  enabled  128.4  20000
forwarding  designated
1     gi1/4  enabled  128.5  2000000  disabled
    disabled
1     gi1/5  enabled  128.6  2000000  disabled
    disabled
1     gi1/6  enabled  128.7  2000000  disabled
    disabled
1     gi1/7  enabled  128.8  2000000  disabled
    disabled
2     gi1/0  enabled  128.1  2000000  disabled
    disabled
2     gi1/3  enabled  128.4  20000
forwarding  root
2     gi1/4  enabled  128.5  2000000  disabled
    disabled
2     gi1/5  enabled  128.6  2000000  disabled
    disabled
2     gi1/6  enabled  128.7  2000000  disabled
    disabled
2     gi1/7  enabled  128.8  2000000  disabled
    disabled
INST                  PRIO.
ID    PORT  STATE     NBR       COST       STS
         ROLE
-------------------------------------------------------------------
1     po1   enabled  128.1000  10000
forwarding  root
1     po2   enabled  128.1001  2000000
disabled    disabled
1     po3   enabled  128.1002  2000000
disabled    disabled
1     po4   enabled  128.1003  2000000
disabled    disabled
2     po1   enabled  128.1000  200000000
blocking    alternate
2     po2   enabled  128.1001  2000000
disabled    disabled
2     po3   enabled  128.1002  2000000
disabled    disabled
2     po4   enabled  128.1003  2000000
disabled    disabled

ENCS-Left#
```

```
 ADDRESS
-------------------------------------------------------------------
1          32768     70:db:98:c3:df:14
70:db:98:c3:df:14
2          61440     f0:b2:e5:56:e4:80
70:db:98:c3:df:14

INST                    PRIO.
ID    PORT    STATE     NBR     COST      STS
      ROLE
-------------------------------------------------------------------
1     gi1/0  enabled  128.1  2000000  disabled
    disabled
1     gi1/3  enabled  128.4  20000
forwarding  designated
1     gi1/4  enabled  128.5  2000000  disabled
    disabled
1     gi1/5  enabled  128.6  2000000  disabled
    disabled
1     gi1/6  enabled  128.7  2000000  disabled
    disabled
1     gi1/7  enabled  128.8  2000000  disabled
    disabled
2     gi1/0  enabled  128.1  2000000  disabled
    disabled
2     gi1/3  enabled  128.4  20000
forwarding  root
2     gi1/4  enabled  128.5  2000000  disabled
    disabled
2     gi1/5  enabled  128.6  2000000  disabled
    disabled
2     gi1/6  enabled  128.7  2000000  disabled
    disabled
2     gi1/7  enabled  128.8  2000000  disabled
    disabled

INST                  PRIO.
ID    PORT  STATE     NBR       COST       STS
         ROLE
-------------------------------------------------------------------
1     po1   enabled  128.1000  10000
forwarding  designated
1     po2   enabled  128.1001  2000000
disabled    disabled
1     po3   enabled  128.1002  2000000
disabled    disabled
1     po4   enabled  128.1003  2000000
disabled    disabled
2     po1   enabled  128.1000  200000000
forwarding  designated
2     po2   enabled  128.1001  2000000
disabled    disabled
2     po3   enabled  128.1002  2000000
disabled    disabled
2     po4   enabled  128.1003  2000000
disabled    disabled
ENCS-Right#
```

From the above summary output, MST instances indicates ID and associated VLAN, and then displays all interfaces as part of VLAN instances. This behaviour differs from the way MST instances are displayed on other Cisco switching platforms.

External Switch MST Configuration

Note It is recommended that VLAN 996-998 is not allowed through the interfaces connecting to ENCS-Left and ENCS-Right. As a result, the external switch MSTP does not participate for VLAN 996-998.

Table 58:

`vlan 100,126-128`
`!`
`spanning-tree mode mst`
`spanning-tree extend system-id`
`spanning-tree uplinkfast`
`!`
`spanning-tree mst configuration`
` name mst_LAN`
` instance 1 vlan 996-998`
` instance 2 vlan 100, 126-128`
`!`
`interface GigabitEthernet1/0/1`
` switchport trunk allowed vlan 100,126-128`
` switchport mode trunk`
`!`
`interface GigabitEthernet1/0/2`
` switchport trunk allowed vlan 100,126-128`
` switchport mode trunk`
	VLANs carrying “Traffic between the VNFs” are NOT sent to the External Switch.
	MST instance priority and MST link COST are kept default in the External Switch.
	MST Priority and COST Configuration in ENCS ensure the External switch is the root and the Interfaces in the External switch connecting to ENCS are in Forwarding state.

Note VLANs carrying traffic between VNFs are not used in external switch and not configured in any interface.

```
Switch#show spanning-tree mst detail

##### MST0    vlans mapped:   1-99,101-125,129-995,999-4094
Bridge        address f0b2.e556.e480  priority      32768 (32768 sysid 0)
Root          address 70db.98c3.df14  priority      32768 (32768 sysid 0)
              port    Gi1/0/2         path cost     0
Regional Root address 70db.98c3.df14  priority      32768 (32768 sysid 0)
                                      internal cost 20000     rem hops 19
Operational   hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured    hello time 2 , forward delay 15, max age 20, max hops    20

GigabitEthernet1/0/1 of MST0 is alternate blocking
Port info             port id          128.1  priority    128  cost       20000
Designated root       address 70db.98c3.df14  priority  32768  cost           0
Design. regional root address 70db.98c3.df14  priority  32768  cost       10000
Designated bridge     address 70db.98c3.dfa0  priority  32768  port id    128.4
Timers: message expires in 5 sec, forward delay 0, forward transitions 0
Bpdus sent 27905, received 31061

GigabitEthernet1/0/2 of MST0 is root forwarding
Port info             port id          128.2  priority    128  cost       20000
Designated root       address 70db.98c3.df14  priority  32768  cost           0
Design. regional root address 70db.98c3.df14  priority  32768  cost           0
Designated bridge     address 70db.98c3.df14  priority  32768  port id    128.4
Timers: message expires in 5 sec, forward delay 0, forward transitions 1
Bpdus sent 27904, received 31070
```

```
##### MST2    vlans mapped:   100,126-128
Bridge        address f0b2.e556.e480  priority      32770 (32768 sysid 2)
Root          this switch for MST2

GigabitEthernet1/0/1 of MST2 is designated forwarding
Port info             port id          128.1  priority    128  cost       20000
Designated root       address f0b2.e556.e480  priority  32770  cost           0
Designated bridge     address f0b2.e556.e480  priority  32770  port id    128.1
Timers: message expires in 0 sec, forward delay 0, forward transitions 1
Bpdus (MRecords) sent 27905, received 31061

GigabitEthernet1/0/2 of MST2 is designated forwarding
Port info             port id          128.2  priority    128  cost       20000
Designated root       address f0b2.e556.e480  priority  32770  cost           0
Designated bridge     address f0b2.e556.e480  priority  32770  port id    128.2
Timers: message expires in 0 sec, forward delay 0, forward transitions 1
Bpdus (MRecords) sent 27904, received 31070

Switch#
```

CIMC Secure Overlay Support

Table 59: Feature History

Feature Name	Release Information	Description
CIMC Secure Overlay Support	NFVIS 4.4.1	This feature enables you to recover NFVIS when it becomes unresponsive. CIMC periodically monitors NFVIS health and when NFVIS health check fails, CIMC establishes a secure tunnel to the remote location. You can then log in to CIMC from the remote site to troubleshoot and recover NFVIS.

The CIMC secure overlay feature is a disaster recovery feature to recover an unresponsive NFVIS. NFVIS is managed from a remote location using a secure (IPsec) tunnel. If NFVIS hangs and does not recover by itself, you must reboot or reinstall NFVIS. With this feature, CIMC periodically monitors NFVIS health. CIMC gets NFVIS secure tunnel details when a secure tunnel is set up from NFVIS to the remote location. When NFVIS health check fails, CIMC waits for an hour by default for NFVIS to recover by itself. If NFVIS does not recover, CIMC establishes a secure tunnel to the remote site.

CIMC can access the remote site at the same NFVIS external IP address. At the remote site, you can log in to CIMC and do the necessary troubleshooting to recover NFVIS. After NFVIS is back up, the CIMC secure tunnel is broken and secure tunnel is established between NFVIS and the remote site again.

Note For this feature to work:

- Secure IPSec tunnel should be configured on NFVIS.
- The feature should be enabled in CIMC during deployment.

The following example shows how to enable secure overlay feature on CIMC:

```
ENCS# scope ipsec-tunnel
ENCS/ipsec-tunnel# show detail
    Enable Failover: no
    NFVIS Down Timeout: 3600
    Tunnel Status: NO_TUNNEL
    Seconds to Failover: 0
ENCS/ipsec-tunnel# set ?
  cli                   CLI options
  nfvis-down-timeout    Number of seconds CIMC waits before starting ipsec tunnel to
remote
  tunnel-failover-enable  Enables failover of ipsec tunnel when NFVIS is down

ENCS/ipsec-tunnel# set tunnel-failover-enable yes
ENCS/ipsec-tunnel# set nfvis-down-timeout 3600
ENCS/ipsec-tunnel# commit
ENCS/# show detail
    Enable Failover: yes
    NFVIS Down Timeout: 3600
    Tunnel Status: NO_TUNNEL
    Seconds to Failover: 0
ENCS/ipsec-tunnel# show
Enable Failover NFVIS Down Timeout Tunnel Status    Seconds to Failover
--------------- ------------------ ---------------------- -------------------
 yes             3600               TUNNEL_ESTABLISHED     0
```

The CIMC tunnel status can be one of the following:

- TUNNEL_ESTABLISHED = all ok and remote can login to CIMC
- TUNNEL_CONFIG_MISSING = IPsec config was not transferred to CIMC
- TUNNEL_ERROR = Runtime error
- UNKNOWN_ERROR

CIMC TACACS Support

Table 60: Feature History

Feature Name	Release Information	Description
CIMC TACACS Support	NFVIS 4.2.1	CIMC TACACS configuration support enabled on NFVIS.

TACACS+ is a security protocol 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+ server running, typically, on a UNIX or Windows NT workstation. You must configure a TACACS+ server before you configure the TACACS+ features on your network access server and make them available.

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

In CIMC TACACS 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 has admin privileges when the user logs into the system, and a user with privilege level 9 or higher has all privileges of an operator at the time of login.

Privilege level below 9 has the read-only privileges.

These two are optional arguments. By default admin-priv is 15 and oper-priv is 11.

To configure a TACACS server on CIMC:

```
Server /# scope tacacs
Server /tacacs# set tacacs-server ip-address
Server /tacacs*# set tacacs-key <keystring>
Server /tacacs*# set tacacs-enable yes
Server /tacacs*# set admin-priv 12
Server /tacacs*# set oper-priv 5
Server /tacacs*# commit
Server /tacacs# show detail
```

To verify the TACACS+ server configuration:

```
Server/tacacs# show detail

tacacs Settings:
    Server domain name or IP address:
    Enable tacacs: yes
    shared-secret key: ******
    admin-priv: 14
    oper-priv: 10
```

For more information about this feature, see TACACS+ Server.

LTE PIM Module on Cisco Catalyst 8200 UCPE

The LTE PIM modules provide 4G LTE cellular connectivity on the Cisco Catalyst 8200 UCPE platform. The following LTE PIM modules are supported on the Cisco Catalyst 8200 UCPE platform:

- P-LTE-GB
- P-LTE-IN
- P-LTE-US
- P-LTE-VZ
- P-LTEA-EA
- P-LTEA-LA

Limitations

The following are the limitations when you use LTE PIM modules on Cisco Catalyst 8200 UCPE platform:

- Only one SIM card is supported at a time. If both SIMs are present, SIM in slot 0 is used. To use SIM in slot 1, slot 0 should be empty.
- IPv6 is not supported.

- Only 4G LTE cellular features mentioned in this section are supported on the LTE PIM module on the Cisco Catalyst 8200 UCPE platform.

Remote Access to Cisco Catalyst 8200 UCPE through LTE PIM Module

Remote access to Cisco Catalyst 8200 UCPE platform through LTE PIM module is supported. This requires users to obtain cellular SIM with static IP address from service providers. After the cellular SIM with static IP address is obtained, you can use remote protocol like SSH to remotely access the Cisco Catalyst 8200 UCPE platform.

```
MacBook:~ $ ssh admin@192.168.100.101
admin@192.168.100.101's password:


Cisco Network Function Virtualization Infrastructure Software (NFVIS)

NFVIS Version: 4.4.1-FC2

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 128.168.241.172 using ssh on C8200-NFVIS
C8200-NFVIS#
```

You can also access Cisco Catalyst 8200 UCPE through a portal.

Show Commands

The following commands are used to display LTE PIM module's information.

The following example shows all cellular information using the **show cellular** command:

```
C8200-NFVIS# show cellular
cellular 1/0
 details IPv4 address =     192.168.100.101
 details Default Gateway Address =  192.168.100.102
 details Interface subnet mask =  255.255.255.252
 details Interface name =   int-CELL-1-0
 details Interface Status =  up
 details DNS IP Address =   198.224.173.135
 details Modem Status =     Call_Connected
 hardware Modem Firmware Version =
SWI9X30C_02.33.03.00
 hardware Device Model ID =                                          EM7455
 hardware International Mobile Subscriber Identity(IMSI) =           311480148702353
 hardware International Mobile Equipment Identity (IMEI) =           356129070591075
 hardware Integrated Circuit Card ID (ICCID) =
89148000001470156497
 hardware Mobile Subscriber Intergrated Services Digital Network Number (MSISDN) =  4085550087

 hardware Factory Serial Number (FSN) =                              LF911203620410
 hardware Current Modem Temperature =                                37 deg C
```

```
 hardware PRI SKU ID =                                                    1102526
 hardware PRI Version =                                                   002.079_001
 hardware Carrier =                                                       VERIZON
 hardware OEM PRI Version =                                               000.016
 hardware Modem Status =                                                  Modem Online
 network Current System Time =     2020/11/15/23/35/12
 network Network Selection Mode =  AUTO
 network Current Service Status =  Packet switched
 network Mobile Country Code(MCC) =  311
 network Mobile Network Code(MNC) =  480
 network Current Roaming Status =  HOME
 network Network =                 Verizon
 network Packet Switch domain(PS) state =  Attached
 network EMM State =               Registered
 network EMM Sub state =           Normal-Service
 network EMM Connection State =    2
 network Tracking Area Code (TAC) =  7936
 network Cell ID =                 7962134
 network Network MTU =             1428
 radio-details Radio Power Mode =           online
 radio-details Radio Access Technology(RAT) Preference =  AUTO
 radio-details Radio Access Technology(RAT) Selected =  LTE
 radio-details LTE RX Channel Number =      2075
 radio-details LTE TX Channel Number =      20075
 radio-details LTE Band =                   4
 radio-details LTE Bandwidth =              15 MHz
 radio-details Current RSSI =               -73 dBm
 radio-details Current RSRP =               -103 dBm
 radio-details Current RSRQ =               -10 dB
 radio-details Current SNR =                5 dB
 radio-details PCI =                        169
Profile 3, Packet Session Status = ACTIVE
Call end mode = None
Session disconnect reason type = None(0)
Session disconnect reason = None(0)
PDN 0
Back off timer is NOT running
Back off error count = 0
Back off timer index = 0
Back off timer array (in minutes) = 0 1 1 1 5 10 15 30 60
Cellular interface back-off is not enforced
Period of back-off = 0 minute(s)
 profile-details 1
  PDP-Type       IPv4v6
  APN-Name       ims
  state          INACTIVE
  PDP-Address    0.0.0.0
  Authentication None
  Username
  Password
 profile-details 2
  PDP-Type       IPv4v6
  APN-Name       vzwadmin
  state          INACTIVE
  PDP-Address    0.0.0.0
  Authentication None
  Username
  Password
 profile-details 3
  PDP-Type       IPv4v6
  APN-Name       we01.VZWSTATIC
  state          ACTIVE
  PDP-Address    192.168.100.101
  Authentication None
```

```
  Username
  Password
 profile-details 4
  PDP-Type       IPv4v6
  APN-Name       vzwapp
  state          INACTIVE
  PDP-Address    0.0.0.0
  Authentication None
  Username
  Password
 profile-details 5
  PDP-Type       IPv4v6
  APN-Name       vzw800
  state          INACTIVE
  PDP-Address    0.0.0.0
  Authentication None
  Username
  Password
 profile-details 6
  PDP-Type       IPv4v6
  APN-Name       vzwclass6
  state          INACTIVE
  PDP-Address    0.0.0.0
  Authentication None
  Username
  Password
 status Firmware Upgrade Status =  "Not Initiated"
 status DM Logging Status =  Stopped
SIM
ID   SIM STATUS
-------------------
0    OK
1    Not inserted

C8200-NFVIS#
```

The following examples shows the individual cellular information using the **show cellular 1/0 [connection | details | hardware | network | profile-details | radio-details | status]** command:

connection : Cellular connection information

```
C8200-NFVIS# show cellular 1/0 connection
Profile 3, Packet Session Status = ACTIVE
Call end mode = None
Session disconnect reason type = None(0)
Session disconnect reason = None(0)
PDN 0
Back off timer is NOT running
Back off error count = 0
Back off timer index = 0
Back off timer array (in minutes) = 0 1 1 1 5 10 15 30 60
Cellular interface back-off is not enforced
Period of back-off = 0 minute(s)
C8200-NFVIS#
```

details : Cellular basic information

```
C8200-NFVIS# show cellular 1/0 details
details IPv4 address =     192.168.100.101
details Default Gateway Address =  192.168.100.102
details Interface subnet mask =  255.255.255.252
details Interface name =   int-CELL-1-0
details Interface Status =  up
```

```
details DNS IP Address =   198.224.173.135
details Modem Status =     Call_Connected
C8200-NFVIS#
```

hardware : Cellular hardware information

```
C8200-NFVIS# show cellular 1/0 hardware
hardware Modem Firmware Version =                                        SWI9X30C_02.33.03.00
hardware Device Model ID =                                                   EM7455
hardware International Mobile Subscriber Identity(IMSI) =                    311480148702353
hardware International Mobile Equipment Identity (IMEI) =                    356129070591075
hardware Integrated Circuit Card ID (ICCID) =                               89148000001470156497
hardware Mobile Subscriber Intergrated Services Digital Network Number (MSISDN) =  4085550087
hardware Factory Serial Number (FSN) =                                       LF911203620410
hardware Current Modem Temperature =                                         37 deg C
hardware PRI SKU ID =                                                        1102526
hardware PRI Version =                                                       002.079_001
hardware Carrier =                                                           VERIZON
hardware OEM PRI Version =                                                   000.016
hardware Modem Status =                                                      Modem Online
C8200-NFVIS#
```

network : Cellular network information

```
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 network
network Current System Time =     2020/11/15/23/39/52
network Network Selection Mode =  AUTO
network Current Service Status =  Packet switched
network Mobile Country Code(MCC) =  311
network Mobile Network Code(MNC) =  480
network Current Roaming Status =  HOME
network Network =                 Verizon
network Packet Switch domain(PS) state =  Attached
network EMM State =               Registered
network EMM Sub state =           Normal-Service
network EMM Connection State =    2
network Tracking Area Code (TAC) =  7936
network Cell ID =                 7962134
network Network MTU =             1428
C8200-NFVIS#
```

profile-details : Cellular profile details

```
C8200-NFVIS# show cellular 1/0 profile-details
profile-details 1
 PDP-Type       IPv4v6
 APN-Name       ims
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
profile-details 2
 PDP-Type       IPv4v6
 APN-Name       vzwadmin
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
```

```
 Password
profile-details 3
 PDP-Type       IPv4v6
 APN-Name       we01.VZWSTATIC
 state          ACTIVE
 PDP-Address    192.168.100.101
 Authentication None
 Username
 Password
profile-details 4
 PDP-Type       IPv4v6
 APN-Name       vzwapp
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
profile-details 5
 PDP-Type       IPv4v6
 APN-Name       vzw800
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
profile-details 6
 PDP-Type       IPv4v6
 APN-Name       vzwclass6
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
C8200-NFVIS#
```

radio-details : Cellular radio details

```
C8200-NFVIS# show cellular 1/0 radio-details
radio-details Radio Power Mode =           online
radio-details Radio Access Technology(RAT) Preference =  AUTO
radio-details Radio Access Technology(RAT) Selected =  LTE
radio-details LTE RX Channel Number =      2075
radio-details LTE TX Channel Number =      20075
radio-details LTE Band =                   4
radio-details LTE Bandwidth =              15 MHz
radio-details Current RSSI =               -72 dBm
radio-details Current RSRP =               -103 dBm
radio-details Current RSRQ =               -17 dB
radio-details Current SNR =                7 dB
radio-details PCI =                        169
C8200-NFVIS#
C8200-NFVIS#
```

status : Cellular status details

```
C8200-NFVIS# show cellular 1/0 status
status Firmware Upgrade Status =  "Not Initiated"
status DM Logging Status =  Stopped
SIM
ID   SIM STATUS
-------------------
0    OK
1    Not inserted
```

```
C8200-NFVIS#
```

The following example shows the Tx and Rx information on the LTE PIM module using the **show bridge-settings cellular-br** command:

```
C8200-NFVIS# show bridge-settings cellular-br
bridge-settings cellular-br
 ip-info interface cellular-br
 ip-info ipv4_address 192.168.100.101
 ip-info netmask 255.255.255.252
 ip-info link-local ipv6 address fe80::7c88:e1ff:de8c:bbc5
 ip-info link-local ipv6 prefixlen 64
 ip-info global ipv6 address ::
 ip-info global ipv6 prefixlen 0
 ip-info mac_address 6f:82:d1:8e:bf:5c
 ip-info mtu  1428
 ip-info txqueuelen 1000
 stats rx_packets 13295
 stats rx_bytes 979420
 stats rx_errors 0
 stats rx_dropped 0
 stats rx_overruns 0
 stats rx_frame 0
 stats tx_packets 11529
 stats tx_bytes 8147515
 stats tx_errors 0
 stats tx_dropped 0
 stats tx_overruns 0
 stats tx_carrier 0
 stats tx_collisions 0
 vlan tag untagged
 dhcp disabled
 dhcp-ipv6 disabled
 slaac-ipv6 disabled
 dpdk-enabled false
C8200-NFVIS#
```

The following example shows how to verify the available PIM module:

```
C8200-UCPE-1N8# show chassis compute inventory pim-modules
POWER
SLOT/BAY/PORT PID TYPE DESCRIPTION STATUS
------------------------------------------------------
0/1/0 P-LTE-US 0x4a10 - on

C8200-UCPE-1N8#
C8200-UCPE-1N8# show chassis
chassis serial FOC241106RG
chassis pid C8200-UCPE-1N8
chassis uuid 4ce1767c-05d0-0000-2903-0f0f000a0b05
chassis bios-version C8200-UCPE_1.0.040920201359
chassis hardware-version V01
chassis compute inventory cpu units CPU0
vendor "Intel(R) Corporation"
family "Pentium 4"
thread-count 8
version "Intel(R) Atom(TM) CPU C3758 @ 2.20GHz"
speed 2200
core-count 8
status "Populated, Enabled"
signature "Type 0, Family 6, Model 95, Stepping 1"
chassis compute inventory memory units DIMM0 "BANK 0"
```

```
capacity 8192
channel-speed 2666
channel-type DDR4
type-detail "Synchronous Registered (Buffered)"
manufacturer "SK Hynix"
serial 434263FD
asset-tag "BANK 0 DIMM0 AssetTag"
part-number HMA81GR7CJR8N-VK
data-width 72
chassis compute inventory memory units DIMM1 "BANK 0"
manufacturer "NO DIMM"
serial "NO DIMM"
asset-tag "NO DIMM"
part-number "NO DIMM"
VENDOR MODEL CAPACITY
-----------------------------------
CISCO eMMC HS-SD/MMC 31.79
ATA SAMSUNG MZ7LH3T8 3840.76

POWER
SLOT/BAY/PORT PID TYPE DESCRIPTION STATUS
-----------------------------------------------------
0/1/0 P-LTE-US 0x4a10 - on

MAX PRODUCT
INPUT OUTPUT ID
------------------------
100 90 -

C8200-UCPE-1N8#
```

Verifying LTE PIM Module Network Connectivity

After Cisco Catalyst 8200 UCPE boots up with a supported LTE PIM and a carrier SIM card, you can verify if the LTE PIM module is connected to the carrier network using the **show cellular 1/0 details** command.

The following example shows how to verify if the LTE PIM module is connected:

```
C8200-NFVIS# show cellular 1/0 details
details IPv4 address =     101.144.109.57  <-- IP address obtained
details Default Gateway Address =  101.144.109.58
details Interface subnet mask =  255.255.255.252
details Interface name =   int-CELL-1-0
details Interface Status =  up
details DNS IP Address =   101.224.173.135
details Modem Status =     Call_Connected  <-- Cellular call connected
C8200-NFVIS#
```

For a successful network connection:

1. The LTE PIM module should be able to obtain an IP address.
    - Example here is 101.144.109.57.
    - If the SIM card has a dynamic IP address, this IP address would change over time. This works the same way as dynamic IP address assignment with wired network.
2. Modem status should show as **`Call_Connected`**.
3. Ping to any public IP network should be successful (8.8.8.8).

If any of the above are incorrect, see the troubleshooting section below.

LTE PIM Module LED

For information on LTE PIM module's LED colors and behaviors, see LTE LED Details.

Online Insertion and Removal (OIR) of LTE PIM Module

You can replace the LTE PIM module without affecting the Cisco Catalyst 8200 UCPE platform system operations with the online insertion and removal (OIR) operation. OIR commands are issued before removing and after installing an LTE PIM. When you perform OIR, use an identical or another variation of LTE PIM module to replace the original one.

Command

chassis compute inventory pim-modules units 0/1/0 [reload | start | stop]



Note Removing the LTE PIM module without executing the **chassis compute inventory pim-modules units 0/1/0 stop** command is not supported.

Examples

```
C8200-NFVIS#
C8200-NFVIS# chassis compute inventory pim-modules units 0/1/0 stop
Are you sure you want to stop this module? [no,yes] yes
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 details
details IPv4 address =     0.0.0.0
details Default Gateway Address =  0.0.0.0
details Interface subnet mask =  0.0.0.0
details Interface name =   int-CELL-1-0
details Interface Status =  down
details DNS IP Address =   0.0.0.0
details Modem Status =     Disconnected
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# show chassis | begin SLOT
SLOT/BAY/PORT  PRODUCT ID   SERIAL  REVISION  NUMBER  MAC ADDRESS
------------------------------------------------------------------------
0/2/0          NIM-LTEA-EA  -       0x1       -       00:5d:73:48:e4:80

                                                 POWER
SLOT/BAY/PORT  PID        TYPE    DESCRIPTION  STATUS
-------------------------------------------------------
0/1/0          P-LTEA-EA  0x4710  -            off

       MAX     PRODUCT
INPUT  OUTPUT  ID
------------------------
100    90      -

C8200-NFVIS#


C8200-NFVIS# chassis compute inventory pim-modules units 0/1/0 start
Are you sure you want to start this module? [no,yes] yes
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 details
details IPv4 address =     192.168.100.101
```

```
details Default Gateway Address =  192.168.100.102
details Interface subnet mask =  255.255.255.252
details Interface name =   int-CELL-1-0
details Interface Status =  up
details DNS IP Address =   198.224.173.135
details Modem Status =     Call_Connected
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# show chassis | begin SLOT
SLOT/BAY/PORT  PRODUCT ID   SERIAL  REVISION  NUMBER  MAC ADDRESS
------------------------------------------------------------------------
0/2/0          NIM-LTEA-EA  -       0x1       -       00:5d:73:48:e4:80

                                                 POWER
SLOT/BAY/PORT  PID        TYPE    DESCRIPTION  STATUS
-----------------------------------------------------
0/1/0          P-LTEA-EA  0x4710  -            on

       MAX     PRODUCT
INPUT  OUTPUT  ID
------------------------
100    90      -

C8200-NFVIS#
```

Turning LTE PIM Module's Radio On/Off

The following examples show how to turn the LTE PIM's radio off or on using the **cellular 1/0 modem-radio-off** and **cellular 1/0 modem-radio-on** commands:

```
C8200-NFVIS#
C8200-NFVIS# cellular 1/0 modem-radio-off
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 radio-details
radio-details Radio Power Mode =           offline
radio-details Offline Reason =             None
radio-details Radio Access Technology(RAT) Preference =  AUTO
radio-details Radio Access Technology(RAT) Selected =  AUTO
radio-details Current RSSI =               -128 dBm
radio-details Channel-Number =             0
radio-details Band =                       Unknown
radio-details ECI0 =                       -2 dBm
C8200-NFVIS#


C8200-NFVIS# cellular 1/0 modem-radio-on
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 radio-details
radio-details Radio Power Mode =           online
radio-details Radio Access Technology(RAT) Preference =  AUTO
radio-details Radio Access Technology(RAT) Selected =  LTE
radio-details LTE RX Channel Number =      2075
radio-details LTE TX Channel Number =      20075
radio-details LTE Band =                   4
radio-details LTE Bandwidth =              15 MHz
radio-details Current RSSI =               -76 dBm
radio-details Current RSRP =               -106 dBm
radio-details Current RSRQ =               -15 dB
radio-details Current SNR =                8 dB
radio-details PCI =                        169
C8200-NFVIS#
```

Resetting Modem on the LTE PIM Module

The following example shows how to reset the LTE PIM's modem using the **cellular 1/0 modem-reset** command:

```
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 hardware
hardware Modem Firmware Version =                                        SWI9X30C_02.33.03.00
hardware Device Model ID =                                                   EM7455
hardware International Mobile Subscriber Identity(IMSI) =                    311480148702353
hardware International Mobile Equipment Identity (IMEI) =                    356129070591075
hardware Integrated Circuit Card ID (ICCID) =                               89148000001470156497
hardware Mobile Subscriber Intergrated Services Digital Network Number (MSISDN) =  4083180087
hardware Factory Serial Number (FSN) =                                       LF911203620410
hardware Current Modem Temperature =                                         36 deg C
hardware PRI SKU ID =                                                        1102526
hardware PRI Version =                                                       002.079_001
hardware Carrier =                                                           VERIZON
hardware OEM PRI Version =                                                   000.016
hardware Modem Status =                                                      Modem Online
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# cellular 1/0 modem-reset
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 hardware
hardware Modem Firmware Version =                                            None
hardware Device Model ID =                                                   None
hardware International Mobile Subscriber Identity(IMSI) =                    None
hardware International Mobile Equipment Identity (IMEI) =                    None
hardware Integrated Circuit Card ID (ICCID) =                                None
hardware Mobile Subscriber Intergrated Services Digital Network Number (MSISDN) =  None
hardware Factory Serial Number (FSN) =                                       None
hardware Current Modem Temperature =                                         None
hardware PRI SKU ID =                                                        None
hardware PRI Version =                                                       None
hardware Carrier =                                                           None
hardware OEM PRI Version =                                                   None
hardware Modem Status =                                                      None <-- modem
is down
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 hardware
hardware Modem Firmware Version =                                        SWI9X30C_02.33.03.00
hardware Device Model ID =                                                   EM7455
hardware International Mobile Subscriber Identity(IMSI) =                    311480148702353
hardware International Mobile Equipment Identity (IMEI) =                    356129070591075
hardware Integrated Circuit Card ID (ICCID) =                               89148000001470156497
hardware Mobile Subscriber Intergrated Services Digital Network Number (MSISDN) =  4083180087
hardware Factory Serial Number (FSN) =                                       LF911203620410
hardware Current Modem Temperature =                                         36 deg C
hardware PRI SKU ID =                                                        1102526
hardware PRI Version =                                                       002.079_001
hardware Carrier =                                                           VERIZON
hardware OEM PRI Version =                                                   000.016
hardware Modem Status =                                                      Modem Online <--
 modem back up
C8200-NFVIS#
```

Modifying Profile on the LTE PIM Module

The following examples show how to create and delete LTE PIM's cellular profile using the **cellular profile create [apn-name | authentication | interface-id | pdp-type | profile-id]** and **cellular profile delete [apn-name | authentication | interface-id | pdp-type | profile-id]** commands:

```
C8200-NFVIS# show cellular 1/0 profile-details | begin "profile-details 7"
C8200-NFVIS#   <-- cellular profile does not exist
C8200-NFVIS#
C8200-NFVIS# cellular profile create apn-name test-apn authentication None interface-id 1/0
 pdp-type IPv4 profile-id 7
resp Profile Management Cmd Executed successfully
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 profile-details | begin "profile-details 7"
profile-details 7   <-- profile created
 PDP-Type       IPv4
 APN-Name       test-apn
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# cellular profile delete apn-name test-apn authentication None interface-id 1/0
 pdp-type IPv4 profile-id 7
resp Profile Management Cmd Executed successfully
C8200-NFVIS#
C8200-NFVIS#
C8200-NFVIS# show cellular 1/0 profile-details | begin "profile-details 7"
C8200-NFVIS#   <-- profile deleted
C8200-NFVIS#
```

DM Log Collection on LTE PIM Module

Diagnostic Monitor (DM) is a Qualcomm proprietary protocol. Diagnostic software tools, such as Sierra Wireless SwiLog and Qualcomm QXDM, are based on DM protocol. These tools can be used to capture data transactions between the modem and the network over the RF interface, which makes them useful tools for troubleshooting 3G and 4G data connectivity or performance issues.

To start DM log collection

The following example shows how to start DM log collection using the **dm-log enable** command:

```
C8200-NFVIS# config terminal
Entering configuration mode terminal
C8200-NFVIS(config)# controller cellular 1/0
C8200-NFVIS(config-cellular-1/0)# dm-log enable
C8200-NFVIS(config-cellular-1/0)# commit
Commit complete.
C8200-NFVIS(config-cellular-1/0)# end
C8200-NFVIS#
```



Note DM log rotation is enabled by default.

To stop DM log collection

The following example shows how to stop DM log collection using the **no dm-log enable** command:

```
C8200-NFVIS# config terminal
Entering configuration mode terminal
C8200-NFVIS(config)# controller cellular 1/0
C8200-NFVIS(config-cellular-1/0)# no dm-log enable
C8200-NFVIS(config-cellular-1/0)# commit
Commit complete.
C8200-NFVIS(config-cellular-1/0)# end
C8200-NFVIS#
```

To view DM log collection status

The following example shows how to view the status of DM log collection using the **show cellular 1/0 status** command:

```
C8200-NFVIS# show cellular 1/0 status
status Firmware Upgrade Status =  "Not Initiated"
status DM Logging Status =  Started <-- DM log enabled
SIM
ID   SIM STATUS
-------------------
0    OK
1    Not inserted

C8200-NFVIS#

C8200-NFVIS# show cellular 1/0 status
status Firmware Upgrade Status =  "Not Initiated"
status DM Logging Status =  Stopped <-- DM log not enabled
SIM
ID   SIM STATUS
-------------------
0    OK
1    Not inserted

C8200-NFVIS#
```

To retrieve the collected DM log file

Download the system's tech-support file.

To configure DM log non-default filter

- SCP the DM log filter onto C8200-UCPE-1N8.

  scp <username>@<server_ip>:<dm_log_filter_file> intdatastore:<dm_log_filter_file>

  Example:

  ```
  scp admin@192.19.145.241:/home/gpsmask.sqf intdatastore:gpsmask.sqf
  ```

- Configure specific DM log filter.

  dm-log filter-path /data/intdatastore/uploads/<dm_log_filter_file>



Note To enable DM log collection with DM log filter, need to issue “commit” twice:

1st time: after configure DM log filter

2nd time: after enable DM log collection

Example:

```
C8200-NFVIS# config terminal
  Entering configuration mode terminal
  C8200-NFVIS(config)# controller cellular 1/0
  C8200-NFVIS(config-cellular-1/0)# dm-log filter-path
/data/intdatastore/uploads/gpsmask.sqf
  C8200-NFVIS(config-cellular-1/0)# commit <-- need to execute before enable DM log
  Commit complete.
  C8200-NFVIS(config-cellular-1/0)#
  C8200-NFVIS(config-cellular-1/0)# dm-log enable
  C8200-NFVIS(config-cellular-1/0)# commit <-- need to execute again
  Commit complete.
  C8200-NFVIS(config-cellular-1/0)# end
  C8200-NFVIS#
```

Note When you use an invalid filter or mistype the filter's name, a DM log file of 0 byte size will be created.

To remove DM log filter

The following example shows how to remove DM log filter using the **no dm-log filter-path** command:

```
C8200-NFVIS# config term
Entering configuration mode terminal
C8200-NFVIS(config)# controller cellular 1/0
C8200-NFVIS(config-cellular-1/0)# no dm-log filter-path
C8200-NFVIS(config-cellular-1/0)# commit
Commit complete.
C8200-NFVIS(config-cellular-1/0)# end
C8200-NFVIS#
```

Firmware Upgrade LTE PIM Module's Modem

From time to time, the LTE PIM's modem requires firmware upgrade to obtain fixes for cellular modem's issues. Firmware upgrade on the LTE PIM module can be done with the following procedure.

1. Obtain firmware and/or OEM PRI files and put them into a server the C8200-UCPE-1N8 can SCP from.

Note You can find the latest firmware files at https://www.cisco.com/c/en/us/support/interfaces-modules/lte-wireless-wan-interfaces/series.html

Caution Use only Cisco certified firmware. Using a firmware version not certified by Cisco may impact the wireless service provider network adversely.

2. SCP the firmware or OEM PRI files from the external source to /data/indatastore/uploads folder of the C8200-UCPE-1N8.

   scp <username>@<server_ip>:<fw_file> intdatastore:<fw_file>

   Example:

   ```
   scp admin@192.19.145.241:/fw_folder/WP76xx_02.28.03.01_ATT_002.071_000.spk
   intdatastore:wp7603_02.28.03.01_ATT.spk
   ```

3. Firmware upgrade the LTE PIM's modem.

   cellular 1/0 upgrade-firmware path /data/intdatastore/uploads/



Note

- Firmware upgrade process will take up to 15 minutes. During this time the modem will be unusable. Please do not remove power or reload the C8200-UCPE-1N8 during the firmware upgrade process.
- Firmware upgrade status can be seen with **show cellular 1/0 status** command.

4. Delete the firmware or OEM PRI files from the /data/indatastore/uploads folder of the C8200-UCPE-1N8.

   system file-delete file name /data/intdatastore/uploads/<fw_file>



Note

- There can only be 1 firmware/OEM PRI file in the /data/indatastore/uploads folder for each firmware upgrade process.
- For EM74xx modems' firmware upgrade, only have 1 .CWE and 1 .NVU files in the folder.
- For WP76xx modems' firmware upgrade, only have 1 .SPK file in the folder.
- For EM74xx modems' and WP76xx modems' OEM PRI upgrade, only have 1 .NVU file in the folder.

Enabling Debug to Troubleshooting LTE PIM Module

Enable debugs to troubleshoot cellular issues can be done with the following procedures:

1. Find out the current system logging level.

   show system logging-level

```
C8200-NFVIS# show system logging-level
  system logging-level configuration info
  system logging-level operational warning
  C8200-NFVIS#
```

2. Set logging level for both configuration and operational to debug.

   system set-log level debug logtype all

```
C8200-NFVIS# system set-log level debug logtype all
  C8200-NFVIS#
  C8200-NFVIS# show system logging-level
  system logging-level configuration debug
  system logging-level operational debug
  C8200-NFVIS#
```

3. Enable debug for cellular logging traces.

   cellular debug eng-mode mode enable

```
  C8200-NFVIS#
  C8200-NFVIS# cellular debug eng-mode mode enable
  C8200-NFVIS#
```

4. Disable debug for cellular logging traces.

cellular debug eng-mode mode disable

```
C8200-NFVIS#
  C8200-NFVIS# cellular debug eng-mode mode disable
  C8200-NFVIS#
```

5. Put system logging level back

   system set-log level <critical | debug | error | info | warning> logtype <all | configuration | operational>

```
C8200-NFVIS#
  C8200-NFVIS# system set-log level info logtype configuration
  C8200-NFVIS# system set-log level warning logtype operational
  C8200-NFVIS#
  C8200-NFVIS# show system logging-level
  system logging-level configuration info
  system logging-level operational warning
  C8200-NFVIS#
```

To retrieve the debug log file

Download the system tech-support file.

Troubleshooting

Check the following use cases if the LTE PIM does not have cellular network connectivity:

- Verify that the LTE PIM module is inserted and recognized by the Cisco Catalyst 8200 UCPE. The following example shows how to verify the same using the command **show chassis** command:

```
C8200-NFVIS# show chassis
chassis serial   FGL2352LLC8
chassis pid      C8200-UCPE-1N8
chassis uuid     4c710d27-b960-0000-2903-0f0f000a0b05
chassis bios-version C8200-UCPE_1.01.042820201140
chassis hardware-version V01
chassis compute inventory cpu units CPU0
 vendor       "Intel(R) Corporation"
 family       "Pentium 4"
 thread-count 8
 version      "Intel(R) Atom(TM) CPU C3758 @ 2.20GHz"
 speed        2200
 core-count   8
 status       "Populated, Enabled"
 signature    "Type 0, Family 6, Model 95, Stepping 1"
chassis compute inventory memory units DIMM0 "BANK 0"
 capacity      8192
 channel-speed 2666
 channel-type  DDR4
 type-detail   "Synchronous Registered (Buffered)"
 manufacturer  Smart
 serial        30304754
 asset-tag     "BANK 0 DIMM0 AssetTag"
 part-number   SR5721G812APGDAME2
 data-width    72
chassis compute inventory memory units DIMM1 "BANK 0"
 manufacturer "NO DIMM"
 serial       "NO DIMM"
 asset-tag    "NO DIMM"
 part-number  "NO DIMM"
VENDOR  MODEL            CAPACITY
-----------------------------------
```

```
CISCO   eMMC  HS-SD/MMC  31.79

                                    HARDWARE  PART
SLOT/BAY/PORT  PRODUCT ID   SERIAL  REVISION  NUMBER  MAC ADDRESS
-------------------------------------------------------------------------
0/2/0          NIM-LTEA-EA  -       0x1       -       00:5d:73:48:e4:80

                                                POWER
SLOT/BAY/PORT  PID        TYPE    DESCRIPTION   STATUS
-----------------------------------------------------
0/1/0          P-LTEA-EA  0x4710  -             on   <--- LTE PIM module is recognized
and power on.

       MAX     PRODUCT
INPUT  OUTPUT  ID
------------------------
100    90      -

C8200-NFVIS#
```

If the LTE PIM module is not recognized, try to reinsert the LTE PIM module.

- Verify if the SIM card is detected by the LTE PIM module. The following example shows how to verify the same using the **show cellular 1/0 status** command:

```
C8200-NFVIS# show cellular 1/0 status
status Firmware Upgrade Status =  "Not Initiated"
status DM Logging Status =  Stopped
SIM
ID   SIM STATUS
-------------------
0    OK             <-- SIM card is detected.
1    Not inserted

C8200-NFVIS#
```

If the SIM card is not detected, perform OIR of the LTE PIM module and re-seat the SIM card.

- Verify if the LTE PIM module is attached to the cellular network. The following example shows how to verify the same using the **show cellular 1/0 network** command:

```
C8200-NFVIS# show cellular 1/0 network
network Current System Time =     2020/12/20/6/56/2
network Network Selection Mode =  AUTO
network Current Service Status =  Packet switched
network Mobile Country Code(MCC) =  311
network Mobile Network Code(MNC) =  480
network Current Roaming Status =  HOME
network Network =                 Verizon
network Packet Switch domain(PS) state =  Attached  <-- LTE PIM module is attached to
the Verizon network.
network EMM State =               Registered
network EMM Sub state =           Normal-Service
network EMM Connection State =    2
network Tracking Area Code (TAC) =  7936
network Cell ID =                 7967246
network Network MTU =             1428
C8200-NFVIS#
```

If the LTE PIM module is not attached to the cellular network, continue on to the next bullet.

- Verify the LTE module has cellular signal or reception. The following example shows how to verify the same using the **show cellular 1/0 radio-details** command:

```
C8200-NFVIS# show cellular 1/0 radio-details
radio-details Radio Power Mode =           online
radio-details Radio Access Technology(RAT) Preference =  AUTO
radio-details Radio Access Technology(RAT) Selected =  LTE
radio-details LTE RX Channel Number =      2075
radio-details LTE TX Channel Number =      20075
radio-details LTE Band =                   4
radio-details LTE Bandwidth =              15 MHz
radio-details Current RSSI =               -71 dBm
radio-details Current RSRP =               -105 dBm
radio-details Current RSRQ =               -11 dB
radio-details Current SNR =                7 dB
radio-details PCI =                        169
C8200-NFVIS#
```

Table 61: Reference Radio Signal Quality

	RSSI (dBm)	RSRP (dBm)	RSRQ (dB)	SNR (dB)
Excellent	> -65	> -84	> -5	> 12.5
Good	-65 to -75	-85 to -102	-9 to -5	10 to 12.5
Fair	-75 to -85	-103 to -111	-12 to -9	7 to 10
Poor	< -85	< -111	< -12	< 7

  - RSSI – Represents the entire received power including the wanted power from the serving cell as well as all co-channel power and other sources of noise.
  - RSRP – The average power received from a single reference signal and its typical range is around -44dbm (good) to -140dbm(bad).
  - RSRQ – Indicates quality of the received signal and its range is typically -19.5dB(bad) to -3dB (good).
  - SNR – The signal-to-noise ratio of the given signal

  If the cellular signal or reception is not good, try to place the cellular antenna at a location where it can pick up good signal or reception.

- Verify the LTE module has correct cellular profile configuration. The following example shows how to verify the same using the **show cellular 1/0 profile-details** command:

```
C8200-NFVIS# show cellular 1/0 profile-details
profile-details 1
 PDP-Type       IPv4v6
 APN-Name       ims
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
profile-details 2
 PDP-Type       IPv4v6
```

```
 APN-Name       vzwadmin
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
profile-details 3
 PDP-Type       IPv4v6
 APN-Name       we01.VZWSTATIC
 state          ACTIVE
 PDP-Address    101.144.109.57
 Authentication None
 Username
 Password
profile-details 4
 PDP-Type       IPv4v6
 APN-Name       vzwapp
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
profile-details 5
 PDP-Type       IPv4v6
 APN-Name       vzw800
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
profile-details 6
 PDP-Type       IPv4v6
 APN-Name       vzwclass6
 state          INACTIVE
 PDP-Address    0.0.0.0
 Authentication None
 Username
 Password
C8200-NFVIS#
```

Before you configure, check with your cellular network carrier for the correct values.

- Verify cellular back-off feature is running or not. When this feature is running, it means that the LTE PIM module is performing necessary back-off operations in order to work smoothly with the provider network. The network provider side initiates this back-off request and the reason for such action depends on the code.

  The following example shows how to verify if the LTE PIM module's cellular back-off feature is running or not, use the **show cellular 1/0 connection** command:

```
C8200-NFVIS# show cellular 1/0 connection
Profile 3, Packet Session Status = ACTIVE
Call end mode = None
Session disconnect reason type = None(0)
Session disconnect reason = None(0)
PDN 0
Back off timer is NOT running  <-- cellular back-off feature is not running
Back off error count = 0
Back off timer index = 0
Back off timer array (in minutes) = 0 1 1 1 5 10 15 30 60
Cellular interface back-off is not enforced
Period of back-off = 0 minute(s)
C8200-NFVIS#
```

If the LTE PIM module has cellular back-off feature running, verify that the cellular profile is configured correctly. If it is configured correctly and you continue to see cellular back-off feature running, check with your cellular network carrier for the reason of back-off.

For more information on cellular back-off feature, see Cellular Back-Off.

- Verify that there is a routing table for LTE PIM module's cellular interface or cellular-br. The following example shows how to verify the same using the **support show router** command:

```
C8200-NFVIS# support show route
default via 101.144.109.58 dev cellular-br metric 10  <--- default route is through the
 cellular-br
10.20.0.0/24 dev int-mgmt-net-br proto kernel scope link src 10.20.0.1
166.144.109.56/30 dev cellular-br proto kernel scope link src 166.144.109.57
192.168.1.0/24 dev lan-br proto kernel scope link src 192.168.1.1
192.168.10.0/24 dev csxbr proto kernel scope link src 192.168.10.11
C8200-NFVIS#
```

If there are multiple default routes, ensure that the cellular-br metric is lower than the other default route.

- Troubleshooting your LTE PIM for cellular network connectivity can also be done through the LEDs located at the front of the LTE PIM module. For more information on the LED behaviors, see 4G LTE-Advanced LEDs.

Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE

Information About Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE

The Cisco Catalyst 8200 UCPE platform includes a slot for a Pluggable Interface Module (PIM) that supports LTE cellular connectivity capabilities. This cellular interface is used to establish control connections and to send VM data traffic and is identified as **cellular-br** for control connections and **cell-data-br** for VM data traffic.

Control Connection via Cellular Interface

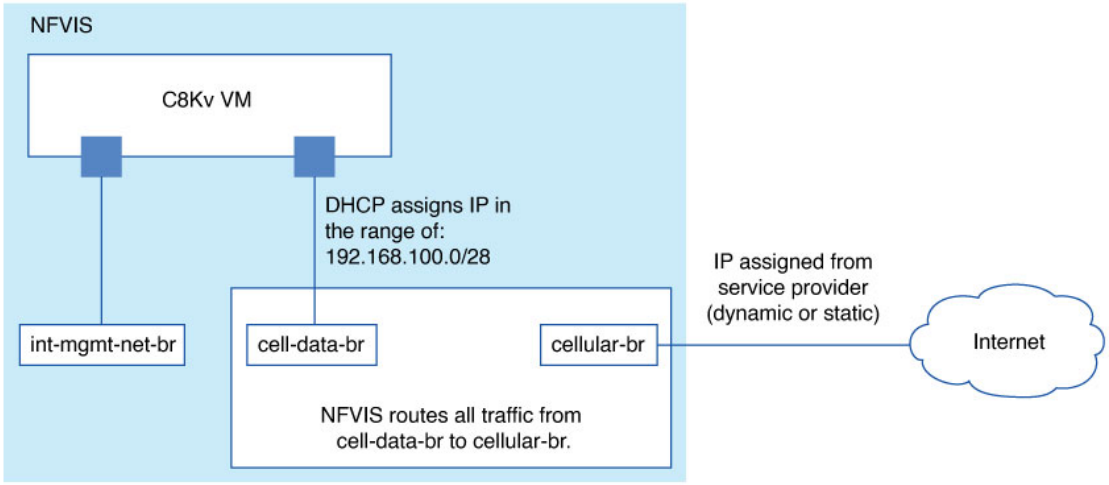
Starting from Cisco Catalyst SD-WAN Manager Release 20.16.1, to establish a control connection between the Cisco Catalyst 8200 UCPE and Cisco SD-WAN Manager via the cellular interface, the **cellular-br** interface must be configured as a VPN interface. When the vDaemon service is initiated through Plug and Play (PnP), NFVIS automatically adds **cellular-br** as a VPN interface. However, if the vDaemon service is started manually using the **hostaction vdaemon-start** command, manually add the **cellular-br** interface under VPN.

VM Data Traffic via Cellular Interface

VMs running on the NFVIS platform can send and receive data through an LTE cellular interface, allowing them to connect to the internet via the cellular network as an alternative to wired connections. This is accomplished by creating a new system-default bridge (**cell-data-br**) and network (**cell-data-net**) on NFVIS, which VMs can use for cellular data transfer.

- **cell-data-net**: This is the network created on NFVIS that VMs can connect to for cellular data transfer. It provides the necessary network configuration for VMs to use the LTE cellular interface for data traffic.
- **cell-data-br**: This is the bridge created on NFVIS that facilitates the connection of VM data traffic to the LTE cellular interface. It acts as the interface through which VMs send and receive data via the cellular network.

Figure 9: Overview of Cellular Interfaces



For VMs deployed on NFVIS to utilize cellular data, they must connect to **cell-data-br**. NFVIS automatically assigns an IP address to the VM's interface using DHCP. Internally, NFVIS reroutes traffic over **cell-data-br**, which is the interface on the NFVIS host that receives its IP address from the cellular service provider.

Benefits of Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE

- Enhanced Connectivity: Enables VMs to use the cellular interface for data transmission, providing an alternative to traditional WAN interfaces.
- Failover Support: Provides a failover mechanism to the cellular interface for control connection only when the WAN interface is down or disconnected.
- Zero Touch Provisioning (ZTP): Allows NFVIS to establish control connections via the cellular interface without manual configuration.

Configure Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE

Control Connections via Cellular

Control connection via the cellular interface (**cellular-br**) are automatically detected and configured by NFVIS.

VM Data Traffic via Cellular

To configure the VM’s interface to use the cellular interface follow these steps:

1. Verify the default bridge/network.

   Use the command **show running-config bridges bridge cell-data-br** to verify that the default bridge and network exist.

2. Configure VM's Network Interface.

   When deploying a VM, configure the VM's network interface to connect to **cell-data-net.** This configuration enables the VM data traffic to be sent via the cellular network.

   For more information about deploying VMs or configuring the VM's network, see VM Life Cycle Management.

Verify Datapath Over LTE Cellular Interface on Cisco Catalyst 8200 UCPE

Verify Control Connections via Cellular

The following is a sample output from the `show control connections` command. This command provides details about the control connections on the Cisco Catalyst 8200 UCPE, specifically highlighting the active control connections between the device and the Cisco SD-WAN Manager.

```
Device# show control connectionsPEER                                                 PEER
                                CONTROLLER
PEER    PEER PEER            SITE       DOMAIN PEER                                  PRIV
  PEER                                     PUB
GROUP
TYPE    PROT SYSTEM IP       ID         ID     PRIVATE IP                            PORT
  PUBLIC IP                                  PORT  ORGANIZATION            LOCAL COLOR
PROXY STATE UPTIME      ID
--------------------------------------------------------------------------------------------------------------
vmanage dtls 192.0.2.22     200        0      192.0.2.29                            12346
 192.0.2.29                              12346 tohofman Org              blue            No
    up     0:00:02:13 0
```

The following is a sample output from the **show cellular 1/0 details** command. This command provides comprehensive information about the cellular interface on the Cisco Catalyst 8200 UCPE, specifically detailing the status, configuration, and operational parameters of the cellular interface identified as 1/0.

```
Device# show cellular 1/0 details
details IPv4 address =     192.0.2.101
details Default Gateway Address =  192.0.2.102
details Interface subnet mask =  255.255.255.240
details Interface name =   int-CELL-1-0
details Interface Status =  up
details DNS IP Address =   203.0.113.135
details Modem Status =     Call_Connected
```

For more information about cellular commands, see LTE PIM Module on Cisco Catalyst 8200 UCPE.

Verify VM Data Traffic via Cellular

The following is a sample output from the **show running-config bridges bridge cell-data-br** command. This command provides current running configuration of the bridge identified as **cell-data-br** on the Cisco Catalyst 8200 UCPE. This bridge facilitates VM data traffic through the LTE cellular interface.

```
Device# show running-config bridges bridge cell-data-br
bridges bridge cell-data-br
 ip address 198.51.100.1 255.255.255.240
```

The following is a sample output from the **support show ifconfig cell-data-br** command. This command provides information about the interface identified as **cell-data-br** on the Cisco Catalyst 8200 UCPE. This interface facilitates VM data traffic through the LTE cellular interface.

```
Device# support show ifconfig cell-data-br
cell-data-br:
flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.0.2.1  netmask 255.255.255.240  broadcast 192.0.2.15
        inet6 fe80::8a5:e0ff:fe18:d141  prefixlen 64  scopeid 0x20<link>
        ether 0a:a5:e0:18:d1:41  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 11  bytes 698 (698.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

External Storage for Cisco ENCS 5400 and Cisco Cloud Services Platforms

Table 62: Feature History

Feature Name	Release Information	Description
Support for External Storage for Cisco Cloud Services Platforms	NFVIS 4.6.1	External disks are supported for Cisco Cloud Services Platforms (CSP).

Restrictions for External Storage for Cisco ENCS 5400 and Cisco Cloud Services Platforms

- A maximum of two RAID groups can be configured for Cisco ENCS 5400 and a maximum of three RAID groups for CSP platforms.

Information about External Storage for Cisco ENCS 5400 and Cisco Cloud Services Platforms

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

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

Note RAID controller is optional on ENCS 5400.

RAID configurations are performed from Cisco IMC for each hardware platform. For UCS-E and CSP devices, all RAID configurations should be performed before installing the NFVIS software. For ENCS 5400, because the OS installation is done on internal SSD, RAID configurations can also be done after the NFVIS software is installed.

This table provides information about the number of RAID groups supported on each platform and the NFVIS release in which the support was introduced:

Platform	Release introduced	Number of RAID groups supported
ENCS 5400	NFVIS 3.8 release	Two RAID groups: First group—extdatastore1 Other group—extdatastore2
Cisco CSP	NFVIS 4.6 release	Three RAID groups: First group—intdatastore (the OS is installed in this group) Second group—extdatastore1 Third group—extdatastore2
UCS-E Single-Wide series	NFVIS 3.5 release	One RAID group for fresh installation
UCS-E Double-Wide series	NFVIS 3.x release and previous releases	One RAID group
	NFVIS 4.1 release	Two RAID groups

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

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

```
nfvis# show system ext-disks

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

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

```
nfvis# show system disk-space

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

Note For more information on the drives and how to create them, see Managing Storage Adapters

Support for 40G Dual Port and Quad-Split NICs on Cisco Cloud Services Platforms

Table 63: Feature History

Feature Name	Release Information	Description
Support for 40G Dual Port and Quad-Split NICs in Cisco Cloud Services Platforms	NFVIS 4.7.1	Starting from this release, the 40G network interface card (NIC) supports dual port modes on Cisco Cloud Services Platform (CSP).

Prerequisites for 40G Dual Port and Quad-Split NICs on Cisco CSP

- Remove the physical network interface card (PNIC) from all networking configurations such as DPDK and SRIOV networks.
- Remove the PNIC from all bridges.
- Remove the PNIC from the port channel.
- Remove the PNIC or virtual functions (VF) of the PNIC from any deployed VM.

Restrictions for 40G Dual Port and Quad-Split NICs on Cisco CSP

- Plug-n-play of the 40G NIC card is not supported.
- The breakout and unbreakout commands should not be executed when a Backup and Restore function is in progress.
- The breakout and unbreakout commands should not be executed when a Factory Default function is in progress.

Note In a NFVIS Backup and Restore, the NIC card configuration mode must be the same on both the source system and the destination system. If the NIC card configuration is different, the restore command is rejected. For more information, see Appendix.

Information About 40G Dual Port and Quad-Split NICs on Cisco CSP

This feature enables the 40G NIC to support two modes on Cisco CSP:

- 2x40 mode – two ports of 40G speed each.
- 4x10 mode – four ports of 10G speed each.

You can convert the 40G NIC from 2x40G mode to 4x10G mode and vice-versa using the breakout commands.

Configure 40G Dual Port and Quad-Split NICs on Cisco CSP Using the CLI

To display the current PNIC modes and adapter information, use the following breakout commands:

```
nfvis# show pnic-breakout

nfvis# show nic
```

To change the PNIC mode from 2x40 to 4x10, use the following breakout command:

```
nfvis# hostaction pnic-breakout device 1 mode 4x10
```

To change the PNIC mode from 4x10 to 2x40, use the following un-breakout command:

```
nfvis# hostaction pnic-breakout device 1 mode 2x40
```

In case of a Return Material Authorization (RMA) of the NIC, you can forcefully breakout or un-breakout between the 2x40G mode and 4x10G mode. For more information on the RMA solution for a 40G NIC, see Appendix.

To forcefully breakout from 2x40 mode to 4x10 mode, use the following command:

```
nfvis# hostaction pnic-breakout force device 1 mode 4x10
```

To forcefully un-breakout from 4x10 mode to 2x40 mode, use the following command:

```
nfvis# hostaction pnic-breakout force device 1 mode 2x40
```

Configuration Examples for 40G Dual Port and Quad-Split NICs on Cisco CSP

1. The following example shows how to change the PNIC mode from 2x40 to 4x10:

```
nfvis# hostaction pnic-breakout device 1 mode 4x10
Warning: Will reboot the system after the mode is changed on the 40G PNIC. All PNIC
configuration like adminstatus, duplex, lldp, promiscuous, speed, sriov, track-state
will be lost and set to default.
Are you sure you want to perform the PNIC breakout?  [no,yes] yes

System message at 2021-06-02 21:15:36...
Commit performed ny via tcp using system.

Broadcast message from root@nfvis (Wed 2021-06-02 21:15:36 UTC):

The system is going down for reboot at Wed 2021-06-02 21:16:36 UTC!
```

2. The following example shows how to change the PNIC mode from 4x10 to 2x40:

```
nfvis# hostaction pnic-breakout device 1 mode 2x40
Warning: Will reboot the system after the mode is changed on the 40G PNIC. All PNIC
configuration like adminstatus, duplex, lldp, promiscuous, speed, sriov, track-state
will be lost and set to default.
Are you sure you want to perform the PNIC breakout?  [no,yes] yes

System message at 2021-06-02 21:15:36...
Commit performed ny via tcp using system.

Broadcast message from root@nfvis (Wed 2021-06-02 21:15:36 UTC):

The system is going down for reboot at Wed 2021-06-02 21:16:36 UTC!
```

3. The following example shows how to forcefully breakout from the 2x40G mode to 4x10G mode:

```
nfvis# hostaction pnic-breakout force device 1 mode 4x10
Warning: Will reboot the system after the mode is changed on the 40G PNIC. All PNIC
configuration like adminstatus, duplex, lldp, promiscuous, speed, sriov, track-state
will be lost and set to default.
Are you sure you want to perform the PNIC breakout?  [no,yes] yes

Broadcast message from root@nfvis (Wed 2021-06-02 21:38:53 UTC):

The system is going down for reboot at Wed 2021-06-02 21:39:53 UTC!
```

4. The following example shows how to forcefully un-breakout from the 4x10 mode to 2x40 mode:

```
nfvis# hostaction pnic-breakout force device 1 mode 2x40
Warning: Will reboot the system after the mode is changed on the 40G PNIC. All PNIC
configuration like adminstatus, duplex, lldp, promiscuous, speed, sriov, track-state
will be lost and set to default.
Are you sure you want to perform the PNIC breakout?  [no,yes] yes

Broadcast message from root@nfvis (Wed 2021-06-02 22:02:40 UTC):

The system is going down for reboot at Wed 2021-06-02 22:03:40 UTC!
```

Verify PNIC Mode Change on a 40G Dual Port and Quad-Split NICs on Cisco CSP

The following example show how to display the current PNIC modes and adapter information:

```
nfvis# show pnic-breakout

DEVNO  PCI  VENDOR  DEVID  ADAPTER                                          MODE      PNICS

-------------------------------------------------------------------------------------------------------
1      5e   8086    1583   Cisco(R) Ethernet Converged NIC XL710-QDA2    2x40  ['eth2-1',
'eth2-2']


nfvis# show nic

SLOTID  ADAPTER                                VENDOR  DEVID  MODE  DEVNO  PNICS

-------------------------------------------------------------------------------------------------------
1       Intel X520 dual port adapter         8086    10fb   NA    NA     ['eth1-1', 'eth1-2']

3       Intel X710-DA4 Quad Port 10Gb SFP+   8086    1572   NA    NA     ['eth3-1', 'eth3-2',
 'eth3-3', 'eth3-4']
        converged NIC

2       Intel XL710-QDA2 Dual Port 40Gb      8086    1583   2x40  1      ['eth2-1', 'eth2-2']

        QSFP converged NIC

5       Intel i350 Quad Port 1Gb Adapter     8086    1521   NA    NA     ['eth5-1', 'eth5-2',
 'eth5-3', 'eth5-4']
4       Intel X520 dual port adapter         8086    10fb   NA    NA     ['eth4-1', 'eth4-2']
```

```
6       Intel X520 dual port adapter        8086    10fb   NA    NA     ['eth6-1', 'eth6-2']
```

C8300-UCPE Virtual RAID and Supported NIMs Specifications

C8300-UCPE Virtual RAID and Supported NIMs Specifications

The C8300-UCPE platform introduces new features that:

- support new Network Interface Modules (NIMs), specifically the C-NIM-8T and C-NIM-4X modules, starting with Cisco NFVIS Release 4.18.
- include a Virtual RAID Controller that supports Software RAID, a technology that combines multiple disk drives into a single unit for improved performance or redundancy. This allows you to create a RAID array using two identical E-SSD-U2N disks, configured as either RAID1 or RAID0.

Prerequisites for RAID controller on C8300-UCPE

Virtual RAID prerequisites

For the C8300-UCPE platform:

- Ensure you have two identical E-SSD-U2N disks installed.
- Enabling Virtual RAID will permanently erase all data on the E-SSD-U2N disks. Therefore, ensure these disks contain no existing data, and back up any data present before proceeding.
- The E-SSD-U2N disks must not have any virtual machines deployed on them. Undeploy any existing virtual machines before enabling Virtual RAID, as their presence will cause enablement to fail.
- If two E-SSD-U2N disks are present during NFVIS installation, RAID1 is automatically created.
- When replacing a disk, use an identical E-SSD-U2N disk.

Restrictions for RAID and NIMs on C8300-UCPE

Restrictions for RAID

These restrictions apply to the RAID feature on C8300-UCPE:

- Once an E-SSD-U2N disk has been enabled as part of a RAID group, it cannot be used as a standalone external disk unless virtual RAID is disabled.
- Only two identical E-SSD-U2N disks are supported in the RAID array; non-identical disks are not supported.

Restrictions for NIMs

The NIM modules on C8300-UCPE do not support:

- L3 mode, which means they cannot perform Layer 3 routing functions.
- MACsec, a security protocol for encrypting data on Ethernet links.

Managing Virtual RAID on C8300-UCPE

This supertask provides an overview and directs you to specific procedures for managing Virtual RAID on the C8300-UCPE. You can configure, disable, or rebuild your Virtual RAID array by following the appropriate subtask

Complete these steps to manage Virtual RAID:

Before you begin

Ensure you have CLI access to the C8300-UCPE

Procedure

Step 1 Enable Virtual RAID. Follow the procedure in Enable Virtual RAID on the C8300-UCPE, on page 331.

Step 2 Disable Virtual RAID. Follow the procedure in Disable Virtual RAID on the C8300-UCPE, on page 332

Step 3 Rebuild Virtual RAID. Follow the procedure in Rebuild Virtual RAID on the C8300-UCPE Platform, on page 332.

What to do next

Verify the status of your Virtual RAID configuration using the **show system:swraid-status** command.

Enable Virtual RAID on the C8300-UCPE

This task guides you through the CLI commands to enable and specify the RAID type for Virtual RAID on the C8300-UCPE, combining multiple disk drives into a single unit for improved performance or redundancy.

Complete these steps to enable Virtual RAID using the CLI:

Before you begin

- Ensure you have two identical E-SSD-U2N disks installed.
- Do not store existing data on the E-SSD-U2N disks.
- Do not deploy any virtual machines on these E-SSD-U2N disks.
- If two E-SSD-U2N disks are present during NFVIS installation, RAID1 is automatically created. To manually enable RAID on the C8300-UCPE, follow the steps provided in the Procedure section.

Procedure

Step 1 Enable Virtual RAID on the C8300-UCPE.

```
nfvis# hostaction swraid-enable
```

The default RAID type is RAID1

Step 2 Specify the RAID type (RAID0 or RAID1) for Virtual RAID.

```
nfvis# hostaction swraid-enable raid-type ?
```

Possible completions include RAID0 and RAID1.

Virtual RAID is enabled on your C8300-UCPE platform with the specified RAID type.

What to do next

- Verify the Virtual RAID configuration. For more information, see Verify Virtual RAID on C8300-UCPE, on page 333

Disable Virtual RAID on the C8300-UCPE

This task provides the CLI command to disable Virtual RAID on the C8300-UCPE. This action is typically performed for maintenance or before re-purposing the drives.

Complete this step to disable Virtual RAID using the CLI:

Before you begin

- Ensure Virtual RAID is currently enabled on the C8300-UCPE.
- Back up any critical data from the Virtual RAID array.

Procedure

Disable Virtual RAID on external disks.

```
nfvis# hostaction swraid-disable
```

Virtual RAID is disabled on your C8300-UCPE.

What to do next

- Verify the Virtual RAID status to confirm it is disabled. For more information, see Verify Virtual RAID on C8300-UCPE, on page 333

Rebuild Virtual RAID on the C8300-UCPE Platform

This task provides the CLI command to rebuild the Virtual RAID array on the C8300-UCPE after a disk replacement. This process restores data redundancy or integrity.

Complete this step to rebuild Virtual RAID using the CLI:

Before you begin

- Ensure that the disk in the Virtual RAID array has been replaced with an identical, new E-SSD-U2N disk..

Procedure

Rebuild Virtual RAID after replacing a faulty disk.

```
nfvis# hostaction swraid-diskreplace
```

The Virtual RAID array on your C8300-UCPE is rebuilt, restoring its intended configuration.

What to do next

- Verify the Virtual RAID status to confirm the rebuild process is complete and successful. For more information, see Verify Virtual RAID on C8300-UCPE, on page 333

Verify Virtual RAID on C8300-UCPE

By verifying the Virtual RAID status, you can confirm the operational status of your RAID array, including its health, level, and disk synchronization.

This section provides the steps to check the Virtual RAID status on the C8300-UCPE.

Before you begin

- Ensure you have CLI access to the C8300-UCPE.

Procedure

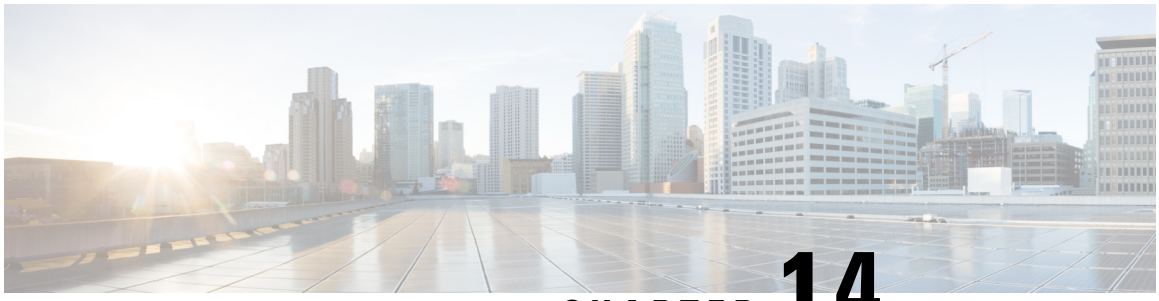
From the CLI interface of the C8300-UCPE, execute the following command:

nfvis# show system:swraid-status

```
swraid-status raid-state Healthy
swraid-status raid-name /dev/md127
swraid-status raid-level 1
swraid-status sync-status True
swraid-status working-devices 2
swraid-status active-devices 2
swraid-status failed-devices 0
swraid-status disk-size 1.8T
swraid-status used-space 32K
DISK NAME           SLOT            STATE
/dev/nvme1n1p1      U2-Upper-Slot   active sync
/dev/nvme2n1p1      U2-Lower-Slot   active sync
```

What to do next

Analyze the output to confirm the Virtual RAID's configuration (level, name, device count), overall health (state, sync status), and the status of individual disks (active, failed).



CHAPTER 14

NFVIS Logging

• Configuring System Logs, on page 335

Configuring System Logs

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

Log entries can be one of the following types:

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

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

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

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

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

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

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

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

Log Rotation

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

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

```
nfvis# logrotate
```

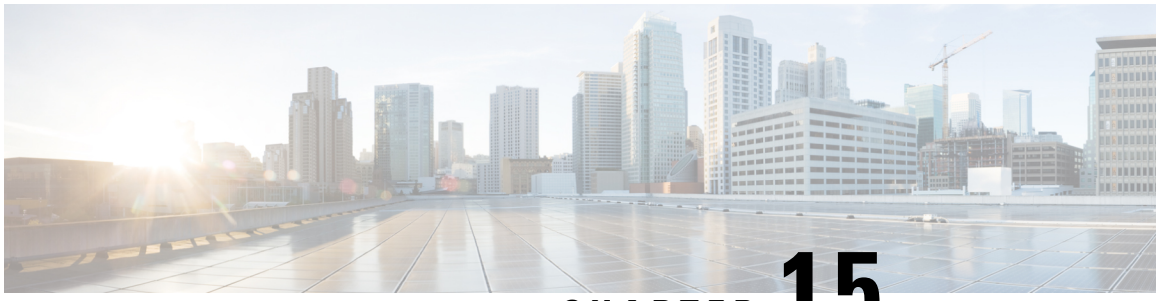
Verifying the System Log Configuration

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

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

System Log APIs and Commands

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



CHAPTER 15

NFVIS Monitoring

- Syslog, on page 337
- NETCONF Event Notifications, on page 339
- SNMP Support on NFVIS, on page 340
- System Monitoring, on page 352

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
```

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 64: Syslog Severity Levels

Severity Level	Description	Numeric Encoding for Severity in the Syslog Message Format
debug	Debug-level messages	7
informational	Informational messages	6
notice	Normal but significant condition	5
warning	Warning conditions	4
error	Error conditions	3
critical	Critical conditions	2
alert	Take action immediately	1
emergency	System is unusable	0



Note By default, the logging severity of syslogs is informational which means all syslogs at informational severity and higher will be logged. Configuring a value for severity will result in syslogs at the configured severity and syslogs which are more severe than the configured severity.

Configure Syslog Facility

The syslog facility can be used to logically separate and store syslog messages on the remote syslog server. For example, syslogs from a particular NFVIS can be assigned a facility of local0 and can be stored and processed in a different directory location on the syslog server. This is useful to separate it from syslogs with a facility of local1 from another device.

To configure syslog facility:

```
configure terminal
system settings logging facility local5
```



Note The logging facility can be changed to a facility from local0 to local7

By default, NFVIS sends syslogs with the facility of local7

Syslog Support APIs and Commands

APIs	Commands
• /api/config/system/settings/logging • /api/operational/system/settings/logging	• system settings logging host • system settings logging severity • system settings logging facility

NETCONF Event Notifications

Cisco Enterprise NFVIS generates event notifications for key events. A NETCONF client can subscribe to these notifications for monitoring the progress of configuration activation and the status change of the system and VMs.

There are two types of event notifications: nfvisEvent and vmlcEvent (VM life cycle event)

To receive event notifications automatically, you can run the NETCONF client, and subscribe to these notifications using the following NETCONF operations:

- --create-subscription=nfvisEvent
- --create-subscription=vmlcEvent

You can view NFVIS and VM life cycle event notifications using the **show notification stream nfvisEvent** and **show notification stream vmlcEvent** commands respectively. For more information see, Event Notifications.

SNMP Support on NFVIS

Introduction about SNMP

Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

The SNMP framework has three parts:

- SNMP manager - The SNMP manager is used to control and monitor the activities of network hosts using SNMP.
- SNMP agent - The SNMP agent is the software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems.
- MIB - The Management Information Base (MIB) is a virtual information storage area for network management information, which consists of collections of managed objects.

A manager can send the agent requests to get and set MIB values. The agent can respond to these requests. Independent of this interaction, the agent can send unsolicited notifications (traps or informs) to the manager to notify the manager of network conditions.

SNMP Operations

SNMP applications perform the following operations to retrieve data, modify SNMP object variables, and send notifications:

- SNMP Get - The SNMP GET operation is performed by a Network Management Server (NMS) to retrieve SNMP object variables.
- SNMP Set - The SNMP SET operation is performed by a Network Management Server (NMS) to modify the value of an object variable.
- SNMP Notifications - A key feature of SNMP is its capability to generate unsolicited notifications from an SNMP agent.

SNMP Get

The SNMP GET operation is performed by a Network Management Server (NMS) to retrieve SNMP object variables. There are three types of GET operations:

- GET: Retrieves the exact object instance from the SNMP agent.
- GETNEXT: Retrieves the next object variable, which is a lexicographical successor to the specified variable.
- GETBULK: Retrieves a large amount of object variable data, without the need for repeated GETNEXT operations.

The command for SNMP GET is :

snmpget -v2c -c [community-name] [NFVIS-box-ip] [tag-name, example ifSpeed].[index value]

SNMP Walk

SNMP walk is an SNMP application that uses SNMP GETNEXT requests to query a network entity for a tree of information.

An object identifier (OID) may be given on the command line. This OID specifies which portion of the object identifier space will be searched using GETNEXT requests. All variables in the subtree below the given OID are queried and their values presented to the user.

The command for SNMP walk with SNMP v2 is:

snmpwalk -v2c -c [community-name] [nfvis-box-ip]

```
snmpwalk -v2c -c myUser 172.19.147.115 1.3.6.1.2.1.1
SNMPv2-MIB::sysDescr.0 = STRING: Cisco NFVIS
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.9.12.3.1.3.1291
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (43545580) 5 days, 0:57:35.80
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING:
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 70
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
IF-MIB::ifIndex.1 = INTEGER: 1
IF-MIB::ifIndex.2 = INTEGER: 2
IF-MIB::ifIndex.3 = INTEGER: 3
IF-MIB::ifIndex.4 = INTEGER: 4
IF-MIB::ifIndex.5 = INTEGER: 5
IF-MIB::ifIndex.6 = INTEGER: 6
IF-MIB::ifIndex.7 = INTEGER: 7
IF-MIB::ifIndex.8 = INTEGER: 8
IF-MIB::ifIndex.9 = INTEGER: 9
IF-MIB::ifIndex.10 = INTEGER: 10
IF-MIB::ifIndex.11 = INTEGER: 11
IF-MIB::ifDescr.1 = STRING: GE0-0
IF-MIB::ifDescr.2 = STRING: GE0-1
IF-MIB::ifDescr.3 = STRING: MGMT
IF-MIB::ifDescr.4 = STRING: gigabitEthernet1/0
IF-MIB::ifDescr.5 = STRING: gigabitEthernet1/1
IF-MIB::ifDescr.6 = STRING: gigabitEthernet1/2
IF-MIB::ifDescr.7 = STRING: gigabitEthernet1/3
IF-MIB::ifDescr.8 = STRING: gigabitEthernet1/4
IF-MIB::ifDescr.9 = STRING: gigabitEthernet1/5
IF-MIB::ifDescr.10 = STRING: gigabitEthernet1/6
IF-MIB::ifDescr.11 = STRING: gigabitEthernet1/7
...
SNMPv2-SMI::mib-2.47.1.1.1.1.2.0 = STRING: "Cisco NFVIS"
SNMPv2-SMI::mib-2.47.1.1.1.1.3.0 = OID: SNMPv2-SMI::enterprises.9.1.1836
SNMPv2-SMI::mib-2.47.1.1.1.1.4.0 = INTEGER: 0
SNMPv2-SMI::mib-2.47.1.1.1.1.5.0 = INTEGER: 3
SNMPv2-SMI::mib-2.47.1.1.1.1.6.0 = INTEGER: -1
SNMPv2-SMI::mib-2.47.1.1.1.1.7.0 = STRING: "ENCS5412/K9"
SNMPv2-SMI::mib-2.47.1.1.1.1.8.0 = STRING: "M3"
SNMPv2-SMI::mib-2.47.1.1.1.1.9.0 = ""
SNMPv2-SMI::mib-2.47.1.1.1.1.10.0 = STRING: "3.7.0-817"
SNMPv2-SMI::mib-2.47.1.1.1.1.11.0 = STRING: "FGL203012P2"
SNMPv2-SMI::mib-2.47.1.1.1.1.12.0 = STRING: "Cisco Systems, Inc."
SNMPv2-SMI::mib-2.47.1.1.1.1.13.0 = ""
...
```

The following is a sample configuration of SNMP walk with SNMP v3:

```
snmpwalk -v 3 -u user3 -a sha -A changePassphrase -x aes -X changePassphrase -l authPriv
-n snmp 172.16.1.101 system
SNMPv2-MIB::sysDescr.0 = STRING: Cisco ENCS 5412, 12-core Intel, 8 GB, 8-port PoE LAN, 2
HDD, Network Compute System
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.9.1.2377
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (16944068) 1 day, 23:04:00.68
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING:
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 70
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
```

SNMP Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. Unsolicited (asynchronous) notifications can be generated as traps or inform requests. Traps are messages alerting the SNMP manager to a condition on the network. Inform requests (informs) are traps that include a request for confirmation of receipt from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.



Note Starting from Release 3.8.1 NFVIS has SNMP Trap support for switch interfaces. If a trap server is setup in the NFVIS snmp configuration, it will send trap messages for both NFVIS and switch interfaces. Both the interfaces are triggered by the link state up or down by unplugging a cable or setting admin_state up or down when a cable is connected.

SNMP Versions

Cisco enterprise NFVIS supports the following versions of SNMP:

- SNMP v1—The Simple Network Management Protocol: A Full Internet Standard, defined in RFC 1157. (RFC 1157 replaces the earlier versions that were published as RFC 1067 and RFC 1098.) Security is based on community strings.
- SNMP v2c—The community-string based Administrative Framework for SNMPv2. SNMPv2c (the "c" stands for "community") is an Experimental Internet Protocol defined in RFC 1901, RFC 1905, and RFC 1906. SNMPv2c is an update of the protocol operations and data types of SNMPv2p (SNMPv2 Classic), and uses the community-based security model of SNMPv1.
- SNMPv3—Version 3 of SNMP. SNMPv3 is an interoperable standards-based protocol defined in RFCs 3413 to 3415. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network.

The security features provided in SNMPv3 are as follows:

- Message integrity—Ensuring that a packet has not been tampered with in transit.
- Authentication—Determining that the message is from a valid source.
- Encryption—Scrambling the contents of a packet to prevent it from being learned by an unauthorized source.

Both SNMP v1 and SNMP v2c use a community-based form of security. The community of managers able to access the agent MIB is defined by an IP address Access Control List and password.

SNMPv3 is a security model in which an authentication strategy is set up for a user and the group in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

Authentication of the community with the user configuration is implemented even though SNMP v1 and v2 traditionally do not require a user configuration to be set. For both SNMP v1 and v2 on NFVIS, the user must be set with the same name and version as the corresponding community name. The user group must also match an existing group with the same SNMP version for snmpwalk commands to work.

SNMP MIB Support

Table 65: 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.
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

System Monitoring

NFVIS provides system monitoring commands and APIs to monitor the host and the VMs deployed on NFVIS. These commands are useful to collect statistics on CPU utilization, memory, disk and ports. The metrics

related to these resources are collected periodically and displayed for a specified duration. For larger durations average values are displayed.

System monitoring enables the user to view historical data on the system's operation.These metrics are also shown as graphs on the portal.

Collection of System Monitoring Statistics

System monitoring statistics are displayed for the requested duration. The default duration is five minutes.

The supported duration values are 1min, 5min, 15min, 30min, 1h, 1H, 6h, 6H, 1d, 1D, 5d, 5D, 30d, 30D with min as minutes, h and H as hours, d and D as days.



Note When a pNIC is actively connected to a vNIC through SRIOV connection, the port usage metrics are displayed only for the last 5 minutes (last 30 values) irrespective of the time interval provided in the CLI to view the port usage.

Example

The following is a sample output of system monitoring statistics:

```
nfvis# show system-monitoring host cpu stats cpu-usage 1h state non-idle
system-monitoring host cpu stats cpu-usage 1h state non-idle
  collect-start-date-time  2019-12-20T11:27:20-00:00
  collect-interval-seconds 10
  cpu
     id               0
     usage-percentage "[7.67, 5.52, 4.89, 5.77, 5.03, 5.93, 10.07, 5.49,
     ...
```

The time at which the data collection started is displayed as **collect-start-date-time**.

The sampling interval at which data is collected is shown as **collect-interval-seconds**.

The data for the requested metric like host CPU statistics is displayed as an array. The first data point in the array was collected at the specified **collect-start-date-time** and each subsequent value at an interval specified by **collect-interval-seconds**.

In the sample output, CPU id 0 has a utilization of 7.67% on 2019-12-20 at 11:27:20 as specified by **collect-start-date-time**. 10 seconds later, it had a utilization of 5.52% since the **collect-interval-seconds** is 10. The third value of cpu-utilization is 4.89% at 10 seconds after the second value of 5.52% and so on.

The sampling interval shown as **collect-interval-seconds** changes based on the specified duration. For higher durations, the collected statistics are averaged at a higher interval to keep the number of results reasonable.

Host System Monitoring

NFVIS provides system monitoring commands and APIs to monitor the host’s CPU utilization, memory, disk and ports.

Monitoring the Host CPU Usage

The percentage of time spent by the CPU in various states, such as executing user code, executing system code, waiting for IO operations, etc. is displayed for the specified duration.

cpu-state	Description
non-idle	100 – idle-cpu-percentage
interrupt	Indicates the percentage of the processor time spent in servicing interrupts
nice	The nice CPU state is a subset of the user state and shows the CPU time used by processes that have a lower priority than other tasks.
system	The system CPU state shows the amount of CPU time used by the kernel.
user	The user CPU state shows CPU time used by user space processes
wait	Idle time while waiting for an I/O operation to complete

The non-idle state is what the user usually needs to monitor. Use the following CLI or API for monitoring CPU usage:

```
nfvis# show system-monitoring host cpu stats cpu-usage <duration> state <cpu-state>
```

```
/api/operational/system-monitoring/host/cpu/stats/cpu-usage/<duration>,<cpu-state>?deep
```

The data is also available in an aggregate form for the minimum, maximum, and average CPU utilization using the following CLI and API:

```
nfvis# show system-monitoring host cpu table cpu-usage <duration>
```

```
/api/operational/system-monitoring/host/cpu/table/cpu-usage/<duration>?deep
```

Monitoring the Host Port Statistics

The statistics collection for non-switch ports is handled by the collectd daemon on all platforms. The input and output rate calculation per port is enabled and the rate calculations are done by the collectd daemon.

Use the **show system-monitoring host port stats** command to display the outputs of the calculations done by collectd for packets/sec, errors/sec and now kilobits/sec. Use the **system-monitoring host port table** command to display the outputs of the collectd stats average for last 5 minutess for packets/sec and kilobits/sec values.

Monitoring Host Memory

Statistics for the physical memory utilization are displayed for the following categories:

Field	Description
buffered-MB	Memory used for buffering I/O
cached-MB	Memory used for caching file system access

Field	Description
free-MB	Memory available for use
used-MB	Memory in use by the system
slab-recl-MB	Memory used for SLAB-allocation of kernel objects, that can be reclaimed
slab-unrecl-MB	Memory used for SLAB-allocation of kernel objects, that can't be reclaimed

Use the following CLI or API for monitoring host memory:

```
nfvis# show system-monitoring host memory stats mem-usage <duration>
```

```
/api/operational/system-monitoring/host/memory/stats/mem-usage/<duration>?deep
```

The data is also available in an aggregate form for the minimum, maximum, and average memory utilization using the following CLI and API:

```
nfvis# show system-monitoring host memory table mem-usage <duration>
```

```
/api/operational/system-monitoring/host/memory/table/mem-usage/<duration>?deep
```

Monitoring Host Disks

Statistics for disk operations and disk space can be obtained for the list of disks and disk partitions on the NFVIS host.

Monitoring Host Disks Operations

The following disk performance statistics are displayed for each disk and disk partition:

Field	Description
io-time-ms	Average time spent doing I/O operations in milliseconds
io-time-weighted-ms	Measure of both I/O completion time and the backlog that may be accumulating
merged-reads-per-sec	The number of read operations that could be merged into already queued operations, that is one physical disk access served two or more logical operations. The higher the merged reads, the better the performance.
merged-writes-per-sec	The number of write operations that could be merged into other already queued operations, that is one physical disk access served two or more logical operations. The higher the merged reads, the better the performance.

Field	Description
bytes-read-per-sec	Bytes read per second
bytes-written-per-sec	Bytes written per second
reads-per-sec	Number of read operations per second
writes-per-sec	Number of write operations per second
time-per-read-ms	The average time a read operation takes to complete
time-per-write-ms	The average time a write operation takes to complete
pending-ops	The queue size of pending I/O operations

Use the following CLI or API for monitoring host disks:

```
nfvis# show system-monitoring host disk stats disk-operations <duration>
```

```
/api/operational/system-monitoring/host/disk/stats/disk-operations/<duration>?deep
```

Monitoring Host Disk Space

The following data related to file system usage, that is how much space on a mounted partition is used and how much is available is collected:

Field	Description
free-GB	Gigabytes available
used-GB	Gigabytes in use
reserved-GB	Gigabytes reserved for the root user

Use the following CLI or API for monitoring host disk space:

```
nfvis# show system-monitoring host disk stats disk-space <duration>
```

```
/api/operational/system-monitoring/host/disk/stats/disk-space/<duration>?deep
```

Monitoring Host Ports

The following statistics for network traffic and errors on interfaces are displayed:

Field	Description
name	Interface name
total-packets-per-sec	Total (received and transmitted) packet rate
rx-packets-per-sec	Packets received per second
tx-packets-per-sec	Packets transmitted per second

Field	Description
total-errors-per-sec	Total (received and transmitted) error rate
rx-errors-per-sec	Error rate for received packets
tx-errors-per-sec	Error rate for transmitted packets

Use the following CLI or API for monitoring host ports:

```
nfvis# show system-monitoring host port stats port-usage <duration>
```

```
/api/operational/system-monitoring/host/port/stats/port-usage/<duration>?deep
```

The data is also available in an aggregate form for the minimum, maximum, and average port utilization using the following CLI and API:

```
nfvis# show system-monitoring host port table
```

```
/api/operational/system-monitoring/host/port/table/port-usage/<duration>,<name>?deep
```

VNF System monitoring

NFVIS provides system monitoring commands and APIs to get statistics on the virtualized guests deployed on NFVIS. These statistics provide data on the VM's CPU utilization, memory, disk and network interfaces.

Monitoring the VNF CPU Usage

The CPU utilization of a VM is displayed for the specified duration using the following fields:

Field	Description
total-percentage	Average CPU utilization across all the logical CPUs used by the VM
id	Logical CPU ID
vcpu-percentage	CPU utilization percentage for the specified logical CPU id

Use the following CLI or API to monitor the CPU usage of the VNF:

```
nfvis# show system-monitoring vnf vcpu stats vcpu-usage <duration>
```

```
/api/operational/system-monitoring/vnf/vcpu/stats/vcpu-usage/<duration>?deep
```

```
/api/operational/system-monitoring/vnf/vcpu/stats/vcpu-usage/<duration>/vnf/<vnf-name>?deep
```

Monitoring VNF memory

The following statistics are collected for VNF memory utilization:

Field	Description
total-MB	Total memory of the VNF in MB
rss-MB	Resident Set Size (RSS) of the VNF in MB The Resident Set Size (RSS) is the portion of memory occupied by a process, that is held in the RAM. The rest of the occupied memory exists in the swap space or file system, because some parts of the occupied memory are paged out, or some parts of the executable are not loaded.

Use the following CLI or API to monitor VNF memory:

```
nfvis# show system-monitoring vnf memory stats mem-usage <duration>
```

```
/api/operational/system-monitoring/vnf/memory/stats/mem-usage/<duration>?deep
```

```
/api/operational/system-monitoring/vnf/memory/stats/mem-usage/<duration>/vnf/<vnf-name>?deep
```

Monitoring VNF Disks

The following disk performance statistics are collected for each disk used by the VM:

Field	Description
bytes-read-per-sec	Bytes read from the disk per second
bytes-written-per-sec	Bytes written to the disk per second
reads-per-sec	Number of read operations per second
writes-per-sec	Number of write operations per second

Use the following CLI or API to monitor VNF disks:

```
nfvis# show system-monitoring vnf disk stats <duration>
```

```
/api/operational/system-monitoring/vnf/disk/stats/disk-operations/<duration>?deep
```

```
/api/operational/system-monitoring/vnf/disk/stats/disk-operations/<duration>/vnf/<vnf-name>?deep
```

Monitoring VNF Ports

The following network interface statistics are collected for VMs deployed on NFVIS:

Field	Description
total-packets-per-sec	Total packets received and transmitted per second
rx-packets-per-sec	Packets received per second

Field	Description
tx-packets-per-sec	Packets transmitted per second
total-errors-per-sec	Total error rate for packet reception and transmission
rx-errors-per-sec	Error rate for receiving packets
tx-errors-per-sec	Error rate for transmitting packets

Use the following CLI or API to monitor VNF ports:

```
nfvis# show system-monitoring vnf port stats port-usage <duration>

/api/operational/system-monitoring/vnf/port/stats/port-usage/<duration>?deep

/api/operational/system-monitoring/vnf/port/stats/port-usage/<duration>/vnf/<vnf-name>?deep
```

ENCS Switch Monitoring

Table 66: Feature History

Feature Name	Release Information	Description
ENCS Switch Monitoring	NFVIS 4.5.1	This feature allows you to calculate the data rate for ENCS switch ports based on the data collected from the ENCS switch.

For ENCS switch ports, the data rate is calculated based on the data collected from the ENCS switch using periodic polling every 10 seconds. Input and output rate in Kbps is calculated based on octets collected from the switch every 10 seconds.

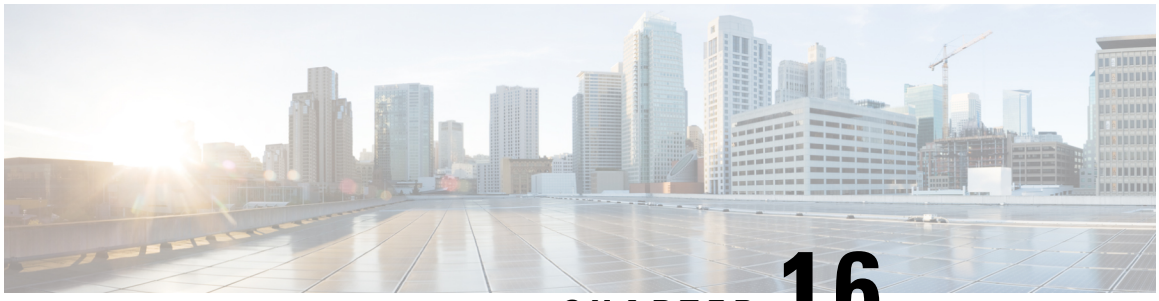
The formula used for the calculation is as follows :

```
Avg rate = (Avg rate – Current interval rate) * (alpha) + Current Interval rate.
Alpha = multiplier/ Scale
Multiplier = scale – (scale * compute_interval)/ Load_interval
Where compute_interval is the polling interval and Load_interval is the interface load
interval = 300 sec and scale = 1024.
```

Because the data is obtained directly from the switch the kbps rate includes Frame Check Sequence (FCS) bytes.

The bandwidth calculation is extended to the ENCS switch port channels using the same formula. Input and output rate in kbps is displayed separately for each gigabit Ethernet port as well as for the corresponding port-channel group the port is associated with.

Use the **show switch interface counters** command to view the data rate calculations.



CHAPTER 16

BIOS Upgrade for C8200 UCPE

Table 67: Feature History

Feature Name	Release Information	Description
BIOS Upgrade for Cisco Catalyst 8200 Series Edge uCPE	Cisco NFVIS Release 4.9.6	This feature outlines the BIOS upgrade process for the Cisco Catalyst 8200 Series Edge uCPE device to resolve the random number generation issue. This upgrade includes updates to Secure Boot FPGA and BIOS capsule.

- Information About BIOS Upgrade for C8200 UCPE, on page 361
- Prerequisites for BIOS Upgrade for C8200 UCPE, on page 362
- Supported Platforms for BIOS Upgrade, on page 362
- Perform BIOS Upgrade for C8200 UCPE, on page 362
- Verify BIOS Upgrade for C8200 UCPE, on page 362

Information About BIOS Upgrade for C8200 UCPE

In NFVIS release 4.9.6, a BIOS upgrade is needed for C8200 UCPE platforms to resolve an issue with entropy functioning, where random numbers are not being generated.

Perform the BIOS upgrade via an SSH session or a serial console using the following commands in sequence:

1. **platform-upgrade fpga secboot**
2. **platform-upgrade bios capsule**

Note The Cisco Catalyst 8200 Series Edge uCPE device reboots after the execution of each command.

Prerequisites for BIOS Upgrade for C8200 UCPE

Ensure that there are no power interruptions during the upgrade process.

Supported Platforms for BIOS Upgrade

Cisco Catalyst 8200 Series Edge uCPE.

Perform BIOS Upgrade for C8200 UCPE

To upgrade the BIOS for the C8200 UCPE, do the following:

Upgrade Secure Boot FPGA

From an SSH terminal or a serial console, execute the **platform-upgrade fpga secboot** command.

The system reboots after the execution of this command.

Update BIOS Capsule

From the SSH terminal or a serial console, execute the **platform-upgrade bios capsule** command .

The system reboots after the execution of this command.

Verify BIOS Upgrade for C8200 UCPE

Verify Secure Boot FPGA Version

The following is a sample output from the **support show aikido-version** command. This command confirms the updated Aikido version is 113.

```
Aikido version: 113
```

Verify BIOS Version

The following is a sample output from the **show platform** command. This command confirms the BIOS has been updated.

Below is a sample output from the **show platform** command, which confirms that the BIOS has been updated. The new BIOS version is highlighted in this example.

```
platform-detail hardware_info Manufacturer "Cisco Systems, Inc."
platform-detail hardware_info PID C8200-UCPE-1N8
platform-detail hardware_info SN FOC241106RR
platform-detail hardware_info hardware-version V01
platform-detail hardware_info UUID 4ce1767c-0660-0000-2903-0f0f000a0b05
platform-detail hardware_info Version 4.9.6-12
platform-detail hardware_info Compile_Time "Wednesday, September 18, 2024 [03:37:38 UTC]"
platform-detail hardware_info CPU_Information "Intel(R) Atom(TM) CPU C3758 @ 2.20GHz 8
cores"
```

```
platform-detail hardware_info Memory_Information "16294400 kB"
platform-detail hardware_info Disk_Size "75.4 GB"
platform-detail hardware_info CIMC_IP NA
platform-detail hardware_info Entity-Name "C8200 UCPE"
platform-detail hardware_info Entity-Desc "Cisco Catalyst 8200 Series Edge uCPE"
platform-detail hardware_info BIOS-Version C8200-UCPE_3.04.073120240748
platform-detail hardware_info CIMC-Version NA
platform-detail software_packages Kernel_Version 3.10.0-1062.4.1.4.el7.x86_64
platform-detail software_packages QEMU_Version 2.12.0
platform-detail software_packages LibVirt_Version 4.5.0
platform-detail software_packages OVS_Version 2.11.4
platform-detail switch_detail UUID NA
platform-detail switch_detail Type NA
platform-detail switch_detail Name NA
platform-detail switch_detail Ports 0
```

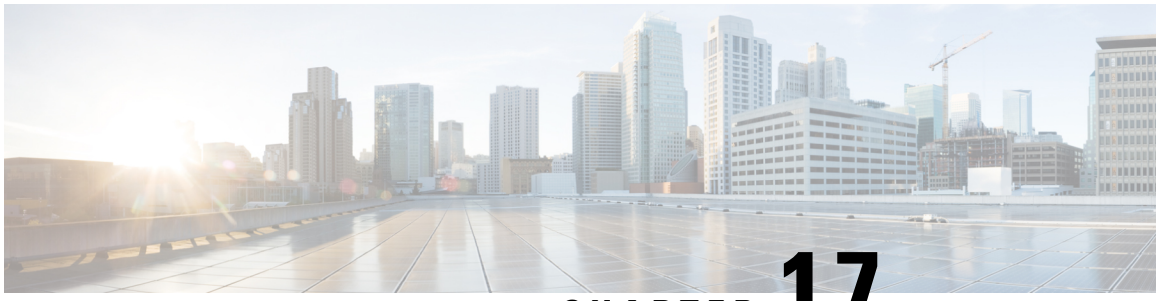
Verify Entropy Functionality

The following is a sample output from the **support show entropy** command. This command verifies whether entropy is functioning correctly.

```
Entropy sources that are available but disabled
1: TPM RNG Device (tpm)
5: NIST Network Entropy Beacon (nist)
9: Qrypt quantum entropy beacon (qrypt)

Available and enabled entropy sources:
0: Hardware RNG Device (hwrng)
2: Intel RDRAND Instruction RNG (rdrand)

Available entropy sources that failed initalization:
6: JITTER Entropy generator (jitter)
```



CHAPTER 17

Troubleshoot and Debug Cisco NFVIS

- Log and Show Commands, on page 365
- SPAN Session or Port Mirroring, on page 367
- Configuring Packet Capture, on page 371

Log and Show Commands

Support Commands and Show Commands

The following commands translate to corresponding linux commands like virsh, ovs and ip:

Command	Description
System	
`show system status`	To display system defaults and services status.
`show system disk-space`	To display information about the system disk space.
`show system memory`	To display information about the system memory. If DPDK is enabled, check if HugePage is available to use.
`show resources cpu-info`	To get information on the resource assignment.
VM	
`support virsh all-info`	To display the output of all supported VM and index by number.
`support virsh dumpxml <num>`	To display all information about one VM index by <num>
`support virsh domiflist <num>`	To display the list of interfaces on VM index by <num> and MAC address of the VNICs.

Command	Description
support flush cache memory	To clear cache memory and free up some system memory for seamless performance of Cisco NFVIS. Clearing caches using support flush cache command can help resolve issues related to outdated or corrupted cache data. For example, clearing the cache in a web browser can help resolve issues such as slow page load times of Cisco NFVIS portal.
Network	
support show ifconfig	To display the configuration details of all network interfaces or a specific interface.
support virsh net-list	To display all the networks in the host
support virsh net-dumpxml <network name>	To display the network information about one network and bridge attachment.
support virsh iface-list	To display a list of interfaces on the host.
Bridge	
support ovs vsctl show: : :	To display an overview of the bridge, port and vlan tag.
support ovs appctl fdb-show <bridge-name>	To display information about the ports of a bridge.
support ovs all-info	To display the output of all supported ovs commands
Firewall	
support show firewall get-all-rule	

Log Files

The tech-support includes all the logs. Download tech-support and record the time of the occurance of error.

Command	Description
show log	To display a list of available log files or content of a specific log file.
show log nfvis_syslog.log	To display syslogs.
show log nfvis_config.log	To display system configuration related logs.
show log esc/escmanager.log	To display VM deployment related logs.
show log switch_confiog.log	To display the built-in switch configuration logs.

SPAN Session or Port Mirroring

About SPAN Sessions

The Switched Port Analyzer (SPAN) or Port Mirroring feature helps you analyze network traffic passing through interfaces or VLANs by using SPAN sessions. The SPAN sessions send a copy (mirror) of the traffic to another interface or VLAN on the switch that has been connected to a network analyzer or monitoring device. SPAN does not affect the switching of network traffic on the source interfaces.



Note You must dedicate a destination port for SPAN use. Except for traffic that is required for the SPAN session, destination ports do not receive or forward traffic. When the SPAN is configured on the system, there might be some performance hit.

SPAN Session Interfaces

The interface can be:

- Physical interface
- LAN SRIOV
- VM's vNIC (virtio net)

In the case of virtio net or SRIOV VF, you have to specify the VM group name and NIC ID of the VM interface. If the VM vNIC is virtio net type, then the SPAN session is applied on the OVS bridge. If VM vNIC is SRIOV VF, then the mirror is applied to the hardware bridge. The interface name is specified for a physical interface, for example, GE0-0 or eth0.

Configuring SPAN Sessions

The SPAN session configuration has the following four parameters:

- Session number—Each SPAN session is identified with a unique number.
- Bridge name—The SPAN session is applied to a bridge. For VLAN mirroring, the bridge must be specified. The bridge name is optional if the source or destination interface is configured for the session.
- Source configuration—The source of the mirror traffic can be one of the following:
  - Packets entering (Rx), or exiting (Tx), or both. You can specify multiple interfaces of any type.
  - You can also specify all interfaces on the OVS bridge.
  - All packets entering a VLAN. You can also specify a list of VLANs.
- Destination configuration—The destination for the mirrored traffic can be one of the following:
  - 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
```

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>

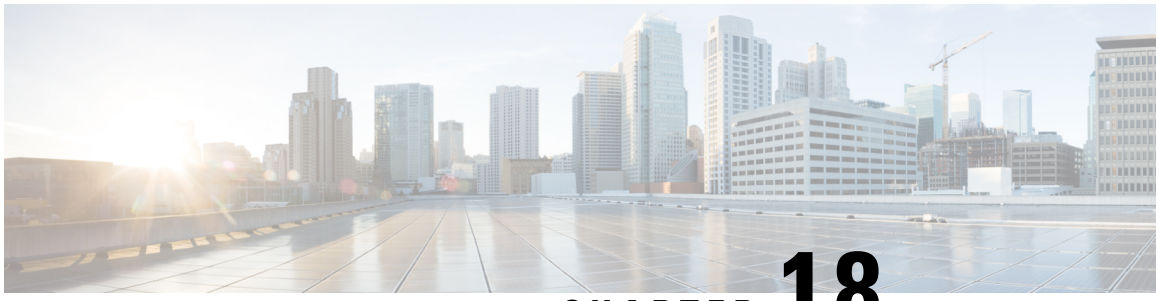
User Name:cisco
Password:**********.   <this is the password you get from step 1 above>
```

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 18

Appendix

- Event Notifications, on page 375
- Syslog Messages, on page 408
- Support for 40G Dual Port and Quad-Split NICs on Cisco Cloud Services Platforms, on page 413
- Granular Role Based Access Control, on page 416

Event Notifications

nfvisEvent

Event Type	Notification Trigger	Notification Output Example
WAN_DHCP_RENEW	DHCP renew operation is performed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T18:06:46.142089+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Wan DHCP IP address is being renewed</status_message> <details>NA</details> <event_type>**WAN_DHCP_RENEW**</event_type> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
BRIDGE_DHCP_RENEW	Bridge DHCP renew operation is performed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2018-04-26T09:47:06.066264+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Bridge DHCP IP address is being renewed</status_message> <details>NA</details> <event_type>**BRIDGE_DHCP_RENEW**</event_type> </nfvisEvent> </notification>
INTF_STATUS_CHANGE	Interface status is changed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T18:12:09.963556+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <event_type>**INTF_STATUS_CHANGE**</event_type> <intf_name>eth7</intf_name> <intf_prv_op>up</intf_prv_op> <intf_op>down</intf_op> <intf_prv_link>down</intf_prv_link> <intf_link>down</intf_link> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
NETWORK_DELETE	A network is deleted.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-09-22T12:42:03.391986+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_message>Network deleted succesfully</status_message> <event_type>**NETWORK_DELETE**</event_type> <network_name>testn1</network_name> <network_bridge/> <network_sriov/> <network_vlan/> <network_trunk/> </nfvisEvent> </notification>
UPGRADE_REGISTER	System upgrade is registered.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T15:57:50.434636+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Upgrade package registration successful: Cisco_NFVIS_Upgrade-3.6.1-698-20170402_042811.nfvispkg</status_message> <event_type>**UPGRADE_REGISTER**</event_type> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
UPGRADE_APPLY	System upgrade is applied.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T16:02:43.885516+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Upgrade Process: In Progress</status_message> <event_type>**UPGRADE_APPLY**</event_type> </nfvisEvent> </notification>
ROTATED_LOGS_DELETE	Rotated logs older than 30 days are deleted by the system.	<?xml version="1.0" encoding="UTF-8"?> <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1"> <ok/> </rpc-reply> <?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T17:38:10.321152+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Deleted rotated logs from archive older than 30 days</status_message> <details>NA</details> <event_type>**ROTATED_LOGS_DELETE**</event_type> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
ROTATED_LOGS_DELETE	Older logs deleted by the system when the total file size of rotated logs exceeds 2GB.	<?xml version="1.0" encoding="UTF-8"?> <rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1"> <ok/> </rpc-reply> <?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-04-26T17:42:10.321152+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>NA</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Rotated logs had exceeded 2G, older logs have been deleted to make space</status_message> <details>NA</details> <event_type>**ROTATED_LOGS_DELETE**</event_type> </nfvisEvent> </notification>
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
OVS_DPDK_SUCCESS	Enable DPDK	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>OVS-DPDK enabled</status_message> <details>NA</details> <event_type>OVS_DPDK_SUCCESS</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
OVS_DPDK_FAILURE	DPDK failure	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Unable to allocate CPU</status_message> <details>NA</details> <event_type>OVS_DPDK_FAILURE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
PNIC_SRIOV_ENABLE	Enable SR-IOV to 2 vfs	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Number of VFs 2</status_message> <details>NA</details> <event_type>PNIC_SRIOV_ENABLE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
PNIC_SRIOV_DISABLE	Disable SR-IOV	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Number of VFs 0</status_message> <details>NA</details> <event_type>PNIC_SRIOV_DISABLE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
PNIC_SRIOV_ENABLE	Enable of disable SR-IOV in error	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Number of VFs 2</status_message> <details>NA</details> <event_type>PNIC_SRIOV_ENABLE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
PNIC_SRIOV_UPDATE	Set switchmode to veb/vepa	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Switchmode vepa</status_message> <details>NA</details> <event_type>PNIC_SRIOV_UPDATE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
PNIC_SRIOV_UPDATE	Set switchmode to veb/vepa in error	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2019-03-15T21:46:28.034133+00:00</eventTime> <nfvisEvent xmlns="http://www.cisco.com/nfvis/notifier"> <user_id>admin</user_id> <config_change>true</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>Physical Interface: eth0-1 Switchmode vepa</status_message> <details>NA</details> <event_type>PNIC_SRIOV_UPDATE</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>
PROMISC_ENABLED	Enable promiscuous mode	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2020-03-27T19:58:52.333682+00:00</eventTime> <nfvisEvent <user_id>admin</user_id> <config_change>false</config_change> <transaction_id>0</transaction_id> <status>SUCCESS</status> <status_code>0</status_code> <status_message>GE0-0: promiscuous mode enabled</status_message> <details>GE0-0: promiscuous mode enabled</details> <event_type>PROMISC_ENABLED</event_type> <severity>INFO</severity> <hostname>NFVIS</hostname> </nfvisEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>
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>

Event Type	Notification Trigger	Notification Output Example
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>

Event Type	Notification Trigger	Notification Output Example
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>
CREATE_FLAVOR	A flavor is created.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:12:29.685+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Flavor creation completed successfully.</status_message> <flavor>ISRv-small</flavor> <vm_source></vm_source> <vm_target></vm_target> <event> <type>**CREATE_FLAVOR**</type> </event> </vmlcEvent> </notification>
DELETE_FLAVOR	A flavor is deleted.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:14:51.425+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Flavor deletion completed successfully.</status_message> <flavor>ISRv-small</flavor> <vm_source></vm_source> <vm_target></vm_target> <event> <type>**DELETE_FLAVOR**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_DEPLOYED	The VM is deployed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:19:16.927+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>VIM Driver: VM successfully created, VM Name: [SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c]</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> <interfaces> <interface> <nicid>0</nicid> <port_id>vnet0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.2</ip_address> <mac_address>52:54:00:31:c5:7f</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <port_id>vnet1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:59:52:41</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> </interfaces> </vm_source> <vm_target></vm_target> <event> <type>**VM_DEPLOYED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_ALIVE	The state of a monitored VM becomes ACTIVE.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:22:47.306+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>VM_Alive event received, VM ID: [SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c]</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> <interfaces> <interface> <nicid>0</nicid> <port_id>vnet0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.2</ip_address> <mac_address>52:54:00:31:c5:7f</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <port_id>vnet1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:59:52:41</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> </interfaces> </vm_source> <vm_target></vm_target> <event> <type>**VM_ALIVE**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_UNDEPLOYED	The VM is undeployed.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:31:40.6+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>204</status_code> <status_message>VIM Driver: VM successfully deleted</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> <interfaces> <interface> <nicid>0</nicid> <port_id>vnet0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.2</ip_address> <mac_address>52:54:00:31:c5:7f</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <port_id>vnet1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:59:52:41</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> </interfaces> </vm_source> <vm_target></vm_target> <event> <type>**VM_UNDEPLOYED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
SERVICE_UPDATED	The VM is updated.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:51:45.5+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Service group update completed successfully</status_message> <depname>1479342258</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <depid>827e871a-30d5-4f5f-a05a-263b7ee3a734</depid> <vm_source></vm_source> <vm_target></vm_target> <event> <type>**SERVICE_UPDATED**</type> </event> </vmlcEvent> </notification>
VM_STOPPED	The VM is stopped per VM action requrest.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:26:05.762+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Successfully stopped VM [SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c].</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_STOPPED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_STARTED	The VM is started per VM action request.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:26:40.398+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Started VM [SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c].</status_message> <depname>1479341445</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>c64d79db-3a29-41a8-8114-c80d42731a5b</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d18dd252-80c8-44f2-ab66-d4481790bb79</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_STARTED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_REBOOTED	The VM is rebooted per VM action request.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T17:36:56.5+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Rebooted VM [SystemAdminTena_ROUTER_0_f17fc494-8535-4b05-b88d-f0fd2effdc7d]</status_message> <depname>1479342258</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>827e871a-30d5-4f5f-a05a-263b7ee3a734</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>d918a3b1-f2a9-4065-9d8e-2135b0a37d87</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_REBOOTED**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_RECOVERY_INIT	A monitored VM is not reachable.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T16:27:51.627+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Recovery event for VM [SystemAdminTena_ROUTER_0_40ae18be-5930-4d94-95ff-dbb0b56ef12b] trigge red. Processing Auto healing. Proceeding with Recovery.</status_message> <depname>1479328919</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>9e7fe4f8-a5f4-4a6d-aad7-121405be4ba4</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>000883fc-77f3-4b9e-aaf6-0f31d88a8f67</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_RECOVERY_INIT**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_RECOVERY_REBOOT	Recovery reboot starts for the monitored VM, which is not reachable.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T16:27:53.979+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>VM [SystemAdminTena_ROUTER_0_40ae18be-5930-4d94-95ff-dbb0b56ef12b] is being rebooted. </statu s_message> <depname>1479328919</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>9e7fe4f8-a5f4-4a6d-aad7-121405be4ba4</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>000883fc-77f3-4b9e-aaf6-0f31d88a8f67</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_RECOVERY_REBOOT**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_RECOVERY_COMPLETE	Recovery reboot completes for the monitored VM, which is not reachable.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-17T16:31:26.934+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Successfully recovered VM [SystemAdminTena_ROUTER_0_40ae18be-5930-4d94-95ff-dbb0b56ef12b]. status_message> <depname>1479328919</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>9e7fe4f8-a5f4-4a6d-aad7-121405be4ba4</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>000883fc-77f3-4b9e-aaf6-0f31d88a8f67</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target> <vmid>000883fc-77f3-4b9e-aaf6-0f31d88a8f67</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> <interfaces> <interface> <nicid>0</nicid> <port_id>vnet0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.2</ip_address> <mac_address>52:54:00:7b:3f:de</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <port_id>vnet1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:96:8a:4d</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> </interfaces> </vm_target> <event> <type>**VM_RECOVERY_COMPLETE**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_MONITOR_UNSET	Monitoring is disabled per VM action request.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-18T13:36:43.613+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Unset monitor completed successfully</status_message> <depname>1479413090</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>742dd335-330c-4bf0-a75d-a44003c645c5</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>23ec3793-37ab-4ec2-a978-a10e08585fdd</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_MONITOR_UNSET**</type> </event> </vmlcEvent> </notification>
VM_MONITOR_SET	Monitoring is enabled per VM action request.	<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2016-11-18T13:40:15.276+00:00</eventTime> <vmlcEvent xmlns="http://www.cisco.com/nfvis/vm_lifecycle"> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Set monitor completed successfully</status_message> <depname>1479413090</depname> <tenant>admin</tenant> <tenant_id>AdminTenantId</tenant_id> <svcid>NULL</svcid> <depid>742dd335-330c-4bf0-a75d-a44003c645c5</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>23ec3793-37ab-4ec2-a978-a10e08585fdd</vmid> <hostid>NFVIS</hostid> <hostname>NFVIS</hostname> </vm_source> <vm_target></vm_target> <event> <type>**VM_MONITOR_SET**</type> </event> </vmlcEvent> </notification>

Event Type	Notification Trigger	Notification Output Example
VM_UPDATED	VM's flavor is changed.	

Event Type	Notification Trigger	Notification Output Example
		<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-12-08T00:50:39.034+00:00</eventTime> <vmlcEvent xmlns='http://www.cisco.com/nfvis/vm_lifecycle'> <status>SUCCESS</status> <status_code>200</status_code> <status_message>VM is resized with flavor [ISRv-medium].</status_message> <user_name>admin</user_name> <depname>1512766000</depname> <tenant>admin</tenant> <tenant_id>adminUUID</tenant_id> <depid>92c11aa1-f6dd-47d1-948f-c8c65b9ef70f</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>1a6f587e-2779-4087-b84d-c0a2c8a481b1</vmid> <vmname>1512766000_ROUTER_0_60d15064-0c6d-49b9-aa4a-80587f626004</vmname> <hostid>NFVIS</hostid> <hostname>nfvis</hostname> <interfaces> <interface> <nicid>0</nicid> <type>virtual</type> <port_id>vnic0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.3</ip_address> <mac_address>52:54:00:3c:ee:5b</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <type>virtual</type> <port_id>vnic1</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:70:06:4a</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> <interface> <nicid>2</nicid> <type>virtual</type> <port_id>vnic2</port_id> <network>lan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:c7:30:1c</mac_address> <netmask>255.255.255.0</netmask> <gateway>192.168.1.1</gateway> </interface> </interfaces> </vm_source> <event> <type>VM_UPDATED</type> </event>

Event Type	Notification Trigger	Notification Output Example
		</vmlcEvent></notification>]]>]]> <?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-12-08T00:50:39.06+00:00</eventTime> <vmlcEvent xmlns='http://www.cisco.com/nfvis/vm_lifecycle'> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Service group update completed successfully</status_message> <user_name>admin</user_name> <depname>1512766000</depname> <tenant>admin</tenant> <tenant_id>adminUUID</tenant_id> <depid>92c11aa1-f6dd-47d1-948f-c8c65b9ef70f</depid> <event> <type>SERVICE_UPDATED</type> </event> </vmlcEvent>

Event Type	Notification Trigger	Notification Output Example
VM_UPDATED	VNIC is added, deleted or updated.	

Event Type	Notification Trigger	Notification Output Example
		<?xml version="1.0" encoding="UTF-8"?> <notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"> <eventTime>2017-12-08T02:10:56.184+00:00</eventTime> <vmlcEvent xmlns='http://www.cisco.com/nfvis/vm_lifecycle'> <status>SUCCESS</status> <status_code>200</status_code> <status_message>Added 1 interface: [managed, net=my-net-1, nicid=3] Updated 2 interface: [managed, net=lan-net, nicid=1],[managed, net=wan-net, nicid=2]</status_message> <user_name>admin</user_name> <depname>1512766000</depname> <tenant>admin</tenant> <tenant_id>adminUUID</tenant_id> <depid>92c11aa1-f6dd-47d1-948f-c8c65b9ef70f</depid> <vm_group>ROUTER</vm_group> <vm_source> <vmid>1a6f587e-2779-4087-b84d-c0a2c8a481b1</vmid> <vmname>1512766000_ROUTER_0_60d15064-0c6d-49b9-aa4a-80587f626004</vmname> <hostid>NFVIS</hostid> <hostname>nfvis</hostname> <interfaces> <interface> <nicid>0</nicid> <type>virtual</type> <port_id>vnic0</port_id> <network>int-mgmt-net</network> <subnet>N/A</subnet> <ip_address>10.20.0.3</ip_address> <mac_address>52:54:00:3c:ee:5b</mac_address> <netmask>255.255.255.0</netmask> <gateway>10.20.0.1</gateway> </interface> <interface> <nicid>1</nicid> <type>virtual</type> <port_id>vnic1</port_id> <network>lan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:70:06:4a</mac_address> <netmask>255.255.255.0</netmask> <gateway>192.168.1.1</gateway> </interface> <interface> <nicid>2</nicid> <type>virtual</type> <port_id>vnic2</port_id> <network>wan-net</network> <subnet>N/A</subnet> <mac_address>52:54:00:c7:30:1c</mac_address> <netmask>255.255.255.0</netmask> <gateway>172.19.181.152</gateway> </interface> <interface> <nicid>3</nicid> <type>virtual</type> <port_id>vnic3</port_id>

Event Type	Notification Trigger	Notification Output Example
		<network>my-net-1</network> <subnet>N/A</subnet> <mac_address>52:54:00:66:b5:c1</mac_address> </interface> </interfaces> </vm_source> <event> <type>VM_UPDATED</type> </event> </vmlcEvent>

Syslog Messages

Event	Trigger Condition	Syslog Messages
NETWORK_CREATE	create a new network	nfvis %SYS-6-NETWORK_CREATE: Network my-net created successfully
NETWORK_UPDATE	modify an existing network	nfvis %SYS-6-NETWORK_UPDATE: Network my-net updated successfully
NETWORK_DELETE	delete a network	nfvis %SYS-6-NETWORK_DELETE: Network my-net deleted successfully
BRIDGE_CREATE	create a new bridge	nfvis %SYS-6-BRIDGE_CREATE: Bridge created successfully: my-bridge
BRIDGE_UPDATE	modify an existing bridge	nfvis %SYS-6-BRIDGE_UPDATE: Updated bridge successfully: my-bridge
BRIDGE_DELETE	delete a bridge	nfvis %SYS-6-BRIDGE_DELETE: Bridge deleted successfully: my-bridge
WAN_DHCP_RENEW	dhcp renew from wan interface	nfvis %SYS-6-WAN_DHCP_RENEW: wan-br DHCP IP address is being renewed
BRIDGE_DHCP_RENEW	bridge dhcp renew	nfvis %SYS-6-BRIDGE_DHCP_RENEW: Bridge DHCP IP address is being renewed
MGMT_DHCP_RENEW	dhcp renew from MGMT interface	nfvis %SYS-6-MGMT_DHCP_RENEW: wan-br DHCP IP address is being renewed
INTF_STATUS_CHANGE	interface status change	nfvis %SYS-6-INTF_STATUS_CHANGE: Interface eth0, changed state to up
UPGRADE_REGISTER	upgrade package registration	nfvis %SYS-6-UPGRADE_REGISTER: Upgrade package registration successful: Cisco_NFVIS_Upgrade-3.6.1-698-20170402_042811.nfvispkg
UPGRADE_APPLY	upgrade process	nfvis %SYS-6-UPGRADE_APPLY: Upgrade Process: In Progress

Event	Trigger Condition	Syslog Messages
RBAC_USER_CREATE	create a new user	nfvis %SYS-6-RBAC_USER_CREATE: Created user admin as administrators successfully
RBAC_USER_PASSWORD_UPDATE	change user's password	nfvis %SYS-6-RBAC_USER_PASSWORD_UPDATE: Set admin password successfully
RBAC_USER_ROLE_UPDATE	change user's role	nfvis %SYS-6-RBAC_USER_ROLE_UPDATE: Modified user: somebody successfully
RBAC_USER_DELETE	delete a user	nfvis %SYS-6-RBAC_USER_DELETE: Deleted rbac user successfully: somebody
RBAC_USERS_INACTIVATED	disable the user	nfvis %SYS-6-RBAC_USERS_INACTIVATED: Following users have been marked as INACTIVE, [user1, user2]. Please take necessary action.
RBAC_USER_ACTIVATED	activate the user	nfvis %SYS-6-RBAC_USER_ACTIVATED: Modified user user1 successfully.
RBAC_PWD_EXPIRED	password expired	nfvis %SYS-6-RBAC_PWD_EXPIRED: User user1's password is older than 60 days. Please reset password.
RBAC_LOGIN_FAILURE	invalid user login	nfvis %SYS-3-RBAC_LOGIN_FAILURE: Login with invalid username from maapi failed
SECURITY_SERVER_CREATE	create server config	nfvis %SYS-6-SECURITY_SERVER_CREATE: TACACS+ server config created successfully.
SECURITY_SERVER_UPDATE	update server config	nfvis %SYS-6-SECURITY_SERVER_UPDATE: TACACS+ server configuration updated successfully.
SECURITY_SERVER_DELETE	delete server config	nfvis %SYS-6-SECURITY_SERVER_DELETE: TACACS+ server deleted successfully.
AAA_TYPE_CREATE	create AAA authentication type	nfvis %SYS-6-AAA_TYPE_CREATE: AAA authentication type TACACS created successfully.
AAA_TYPE_UPDATE	update AAA authentication type	nfvis %SYS-6-AAA_TYPE_UPDATE: AAA authentication type TACACS+ updated successfully. AAA authentication updated to use TACACS+ server
RECREATE_CERTIFICATE	recreate self-sign certificate	nfvis %SYS-6-RECREATE_CERTIFICATE: Self Signed Certificate re-created. Application connection may become temporarily unavailable.
CERT_CSR_CREATE	create a CSR file	nfvis %SYS-6-CERT_CSR_CREATE: signing-request created /data/intdatastore/download/nfvis.csr
CERT_SWITCH_CERT	switch to use different certificate	nfvis %SYS-6-CERT_SWITCH_CERT: switch certificate from ca-signed to self-signed.
CERT_CA_CERT_INSTALL	install CA signed certificate	nfvis %SYS-6-CERT_CA_CERT_INSTALL: ca-signed certificate file:// installed

Event	Trigger Condition	Syslog Messages
REBOOT	system reboot	nfvis %SYS-6-REBOOT: System will be rebooted
SHUTDOWN	system shutdown	nfvis %SYS-6-SHUTDOWN: System will be shutdown
LOGGING_FAILURE	logging failure	nfvis %SYS-6-LOGGING_FAILURE: Unable to write to log file nfvis_config.log. Log message: log_config.CONFIG_LOGGER: File not found.
DISK_SPACE_ALMOST_FULL	disk space almost full	nfvis %SYS-6-DISK_SPACE_ALMOST_FULL: 'lv_data' currently occupies 95% of available disk space, which is more than or equal to the threshold of 90%.
ROTATED_LOGS_DELETE	delete rotated logfiles when accumulated rotated log files reach 2GB	nfvis %SYS-6-ROTATED_LOGS_DELETE: Deleted rotated logs from archive older than 30 days
TIME_UPDATE	Change system time manually	nfvis %SYS-6-TIME_UPDATE: Manual time updated successfully Manual time is now set to 2018-04-26 11:43:00
TIMEZONE_UPDATE	Change system timezone	nfvis %SYS-6-TIMEZONE_UPDATE: Timezone updated successfully. Timezone is now set to US/Eastern
FILE_COPY_STATUS	copy status of file	nfvis %SYS-6-FILE_COPY_STATUS: hostaction.py Copied Successfully.
CREATE_IMAGE	create image	nfvis %SYS-6-CREATE_IMAGE: Image creation successful: TinyLinux.tar.gz
DELETE_IMAGE	delete image	nfvis %SYS-6-DELETE_IMAGE: Image deletion successful: TinyLinux.tar.gz
CREATE_FLAVOR	create flavor	nfvis %SYS-6-CREATE_FLAVOR: Profile creation successful: small
DELETE_FLAVOR	delete flavor	nfvis %SYS-6-DELETE_FLAVOR: Profile deletion successful: small
VM_DEPLOYED	vm deployment	nfvis %SYS-6-VM_DEPLOYED: VM deployment successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_ALIVE	vm alive	nfvis %SYS-6-VM_ALIVE: VM active successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
SERVICE_ALIVE	service alive	nfvis %SYS-6-SERVICE_ALIVE: Service group deployment completed successfully!

Event	Trigger Condition	Syslog Messages
VM_UNDEPLOYED	vm undeployed	nfvis %SYS-6-VM_UNDEPLOYED: VM undeployment successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c SERVICE_UNDEPLOYED service undeployed nfvis %SYS-6-SERVICE_UNDEPLOYED: Service group undeployment completed successfully
VM_UPDATED (update flavor)	vm updated	nfvis %SYS-6-VM_UPDATED: VM update successful: VM is resized with flavor [ISRv-medium].
VM_UPDATED (vnic add / delete / update)	vm updated	nfvis %SYS-6-VM_UPDATED: VM update successful: Added 1 interface: [managed, net=my-net-1, nicid=3] Updated 2 interface: [managed, net=lan-net, nicid=1],[managed, net=wan-net, nicid=2]
SERVICE_UPDATED	service updated	nfvis %SYS-6-SERVICE_UPDATED: Service group update completed successfully
VM_STOPPED	vm stopped	nfvis %SYS-6-VM_STOPPED: VM stop successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_STARTED	vm started	nfvis %SYS-6-VM_STARTED: VM start successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_REBOOTED	vm rebooted	nfvis %SYS-6-VM_REBOOTED: VM reboot successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_RECOVERY_INIT	vm recovery initiation	nfvis %SYS-6-VM_RECOVERY_INIT: VM recovery initiation successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_RECOVERY_REBOOT	vm recovery reboot	nfvis %SYS-6-VM_RECOVERY_REBOOT: VM recovery reboot successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_RECOVERY_COMPLETE	vm recovery complete	nfvis %SYS-6-VM_RECOVERY_COMPLETE: VM recovery successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_MONITOR_UNSET	vm monitoring unset	nfvis %SYS-6-VM_MONITOR_UNSET: Unsetting VM monitoring successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
VM_MONITOR_SET	vm monitoring set	nfvis %SYS-6-VM_MONITOR_SET: Setting VM monitoring successful: SystemAdminTena_ROUTER_0_df6733c1-0768-4ae6-8dce-b223ecdb036c
ROTATED_LOGS_DELETE (When logs older than 30 days are present)	delete rotated logs	nfvis %SYS-6-ROTATED_LOGS_DELETE: Deleted rotated logs from archive older than 30 days

Event	Trigger Condition	Syslog Messages
ROTATED_LOGS_DELETE (When Log file size exceed 2GB, older logs are deleted)	delete rotated logs	nfvis %SYS-6-ROTATED_LOGS_DELETE: Rotated logs had exceeded 2G, older logs have been deleted to make space
CIMC_PASSWORD_UPDATE	cimc password update operation	nfvis %SYS-6-CIMC_PASSWORD_UPDATE: CIMC password change is successful
BIOS_PASSWORD_UPDATE	bios password update operation	nfvis %SYS-6-BIOS_PASSWORD_UPDATE: BIOS password change is successful
SECURE_OVERLAY_CREATING	create secure overlay	nfvis %SYS-6-SECURE_OVERLAY_CREATING: Secure Overlay mgmthub initial creation. Active local bridge: wan-br
SECURE_OVERLAY_UP	secure overlay is up	nfvis %SYS-6-SECURE_OVERLAY_UP: Secure Overlay mgmthub up. Active bridge: wan-br Secure Overlay up after network interuption
SECURE_OVERLAY_DELETE	secure overlay is deleted	nfvis %SYS-6-SECURE_OVERLAY_DELETE: Secure Overlay deleted
SECURE_OVERLAY_ERROR	error in secure overlay	nfvis %SYS-3-SECURE_OVERLAY_ERROR: Secure Overlay mgmthub creation in error. Active bridge: wan-br Secure overlay initial creation nfvis %SYS-3-SECURE_OVERLAY_ERROR: Secure Overlay mgmthub creation in error. Active bridge: wan-br Cannot ping remote system ip address 10.0.0.1
WAN_DHCP_SWITCHOVER	WAN bridge toggle	nfvis %SYS-6-WAN_DHCP_SWITCHOVER: Switch over to bridge wan-br for auto DHCP enablement successful
WAN_DHCP_TOGGLE_END	WAN bridge toggle	nfvis %SYS-6-WAN_DHCP_TOGGLE_END: Disabling bridge toggle for auto DHCP enablement.
ROUTE_DISTRIBUTION_DOWN	Route distribution down	nfvis %SYS-6-ROUTE_DISTRIBUTION_DOWN: Neighbor Address: 172.25.221.106
ROUTE_DISTRIBUTION_START	Route distribution start	nfvis %SYS-6-ROUTE_DISTRIBUTION_START: Route Distribution initial creation. Neighbor Address: 172.25.221.106
ROUTE_DISTRIBUTION_ERROR	Route distribution in error state	nfvis %SYS-3-ROUTE_DISTRIBUTION_ERROR: Neighbor Address: 172.25.221.106
ROUTE_DISTRIBUTION_DELETE	Route distribution deleted	nfvis %SYS-6-ROUTE_DISTRIBUTION_DELETE: All Neighbor Addresses deleted
ROUTE_DISTRIBUTION_UP	Route distribution up	nfvis %SYS-3-ROUTE_DISTRIBUTION_UP: Neighbor Address: 172.25.221.106

Event	Trigger Condition	Syslog Messages
OVS_DPDK_SUCCESS	Enable DPDK	nfvis %SYS-3-OVS_DPDK_SUCCESS: OVS-DPDK enabled
OVS_DPDK_FAILURE	DPDK failure	nfvis %SYS-3-OVS_DPDK_FAILURE: Unable to allocate CP
BACKUP_INIT	Backup configuration initiation	nfvis %SYS-6-BACKUP_INIT: Starting backup: configuration-xxx
BACKUP_SUCCESS	Backup configuration successful	nfvis %SYS-6-BACKUP_SUCCESS: Backup configuration-xxx completed successfully
BACKUP_FAILURE	Backup configuration failure	nfvis %SYS-3-BACKUP_FAILURE: Backup configuration-xxx failed
RESTORE_INIT	Restore initiation	nfvis %SYS-6-RESTORE_INIT: Restore started
RESTORE_ SUCCESS	Successful restore	nfvis %SYS-6-RESTORE_ SUCCESS: Restore successful
RESTORE_FAILURE	Failure to restore	nfvis %SYS-3-RESTORE_FAILURE: Restore failed - internal error

Support for 40G Dual Port and Quad-Split NICs on Cisco Cloud Services Platforms

NFVIS Backup and Restore

In a NFVIS Backup and Restore, the PNIC mode of the source system and the destination system must be the same. If not, the restore command is rejected.

Backup (Source System)	Restore (Destination System)	Result
2x40 mode configuration	2x40 mode configuration	**`nfvis# hostaction restore file-path intdatastore:sample-dir/sample2.bkup`** 2021-06-07 21:51:22,079 INFO [hostaction:/hostaction:hostaction] [restore] Restore completed successfully.

Backup (Source System)	Restore (Destination System)	Result
2x40 mode configuration	4x10 mode configuration	Restore command is rejected. **nfvis# hostaction restore file-path intdatastore:sample-dir/sample4.bkup** Error: Unable to restore - 40G PNIC mode configuration mismatch.
4x10 mode configuration	2x40 mode configuration	Restore command is rejected. **nfvis# hostaction restore file-path intdatastore:sample-dir/sample4.bkup** Error: Unable to restore - 40G PNIC mode configuration mismatch.
4x10 mode configuration	4x10 mode configuration	**nfvis# hostaction restore file-path intdatastore:sample-dir/sample2.bkup** 2021-06-07 21:51:22,079 INFO [hostaction:/hostaction:hostaction] [restore] Restore completed successfully.

Return Material Authorization (RMA)

Note Before powering down the CSP device, ensure to remove the vMedia configuration used for NFVIS installation.

Table 68:

Original Card Mode	New Card Mode	Process
2x40	2x40	• Shut down the CSP. • Insert the new 2x40 card in the same slot as the old RMAed card. • Turn on the CSP. • Confirm that the PNICs are in ok state by checking **show system status**. • Interfaces come up as ethx-1 and ethx-2 where x is the slot. Run the **show running-config pnic** and **show pnic** commands to check PNIC information.

Original Card Mode	New Card Mode	Process
4x10	4x10	• Shut down the CSP. • Insert the new 4x10 card in the same slot as the old RMAed card. • Turn on the CSP. • Confirm that the PNICs are in ok state by checking **`show system status`**. • Interfaces come up as ethx-1-1, ethx-1-2, ethx-1-3, and ethx-1-4, where x is the slot. Run the **`show running-config pnic`** and **`show pnic`** commands to check PNIC information.
2x40: ethx-1 ethx-2	4x10	• Shut down the CSP. • Insert the new 4x10 card in the same slot as the old RMAed card. • Turn on the CSP. • NFVIS reboots again since the PNIC status is not ok. You see the interfaces ethx-1, ethx-2, ethx-1-1, ethx-1-2, ethx-1-3, and ethx-1-4 in the running configuration. Running the **`show system status`** command shows that the PNIC status is not ok.c • Run the **`show pnic-breakout`** or **`show nic`** commands to get the device ID of the 40G NIC card whose PNIC mode needs to be changed. • Use the device ID got from the previous step to run the un-breakout command: **`hostaction pnic-breakout force device x mode 2x40`** (where x is device ID). • System reboots after the command is executed. • Confirm that PNICs are in ok state by running the **`show system status`** command. • Interfaces come up as ethx-1, ethx-2 where x is the slot. Run the **`show running-config pnic`** and **`show pnic`** commands to check PNIC information.

Original Card Mode	New Card Mode	Process
4x10: ethx-1-1 ethx-1-2 ethx-1-3 ethx-1-4	2x40	• Shut down the CSP. • Insert the new 2x40 card in the same slot as the old RMAed card. • Turn on the CSP. • NFVIS reboots again since the PNIC status is not ok. You see the interfaces ethx-1, ethx-2, ethx-1-1, ethx-1-2, ethx-1-3, and ethx-1-4 in running configuration. Running the **`show system status`** command shows that the PNIC status is not ok. Run the **`show running-config pnic`** and **`show pnic`** commands to check PNIC information. • Run the **`show pnic-breakout`** or **`show nic`** commands to get the device ID of the 40G NIC card whose PNIC mode needs to be changed. • Use the device ID got from the previous step to run the breakout command: **`hostaction pnic-breakout force device x mode 2x40`** (where x is device ID). • System reboots after the command is executed. • Confirm that PNICs are in ok state by running the **`show system status`** command. • Interfaces come up as ethx-1-1, ethx-1-2, ethx-1-3, and ethx-1-4 where x is the slot. Run the **`show running-config pnic`** and **`show pnic`** commands to check PNIC information.

Granular Role Based Access Control

Capability Matrix

Table 69:

Operations/Roles	"admin" Super User	Administrator	Operator	Auditor	VNF Group Administrator	VNF Group Operator	VNF Group Auditor
User Management							
Create, update, delete new logins and assign roles	Y	Y	N	N	N	N	N
List all logins	Y	Y	Y	Y	Y	Y	Y

Operations/Roles	"admin" Super User	Administrator	Operator	Auditor	VNF Group Administrator	VNF Group Operator	VNF Group Auditor
Ability to reset password on any login	Y	Y	N	N	Y	N	N
Create and delete custom VNF groups and assign users to VNF groups	Y	Y	N	N	N	N	N
VNF Management							
Create	Y	Y	N	N	Y	N	N
Delete or update VNFs	Y	Y	N	N	Specific VNFs	N	N
Update VNF group name	Y	Y	N	N	N	N	N
Start and stop VNFs	Y	Y	Y	N	Specific VNFs	Specific VNFs	N
VNF snapshots	Y	Y	Y	N	Specific VNFs	Specific VNFs	N
Restore from snapshot	Y	Y	N	N	Specific VNFs	N	N
List all VNF instances	Y	Y	Y	Y	Specific VNFs	Specific VNFs	Specific VNFs
Access the VNF console	Y	Y	Y	N	Specific VNFs	Specific VNFs	N
Assign VNF to a custom group	Y	Y	N	N	N	N	N
Device Level Operations							

Operations/Roles	"admin" Super User	Administrator	Operator	Auditor	VNF Group Administrator	VNF Group Operator	VNF Group Auditor
View and monitor device level statistics, including logs	Y	Y	Y	Y	Y	Y	Y
Factory reset	Y	Y	N	N	N	N	N
System-wide device configurations such as DNS	Y	Y	N	N	N	N	N
Device backup and restore	Y	N	N	N	N	N	N
Platform upgrades	Y	Y	N	N	N	N	N
Change BIOS password, change CIMC password, PNIC-breakout, power cycle, host reboot, and hostaction shutdown	Y	Y	N	N	N	N	N
Clear BIOS spassword, SSH keys, and stats files	Y	Y	N	N	N	N	N
Clear cache, mgmt-dhcp-renew, wan-dhcp-renew, and PnP actions	Y	Y	Y	N	N	N	N

Operations/Roles	"admin" Super User	Administrator	Operator	Auditor	VNF Group Administrator	VNF Group Operator	VNF Group Auditor
System certificate modifications, ISCSI settings, and file delete operation	Y	Y	N	N	N	N	N
Tech support and logs	Y	Y	Y	Only through portal	Y	Y	Only through portal
Create VFs for SR-IOV, and enable DPDK	Y	Y	N	N	N	N	N
Image Repository Changes							
Upload new VNF images	Y	Y	N	N	Y	N	N
Upload new NFVIS upgrade files	Y	Y	N	N	Y	N	N
File transfer to the device (certificates)	Y	Y	N	N	Y	N	N
File transfer from the device	Y	Y	N	N	Y	N	N
File-copy and file-download	Y	Y	N	N	Y	N	N
File-delete	Y	Y	N	N	N	N	N
View contents	Y	Y	Y	Y	Y	Y	Y
Clustering Operations							

Operations/Roles	"admin" Super User	Administrator	Operator	Auditor	VNF Group Administrator	VNF Group Operator	VNF Group Auditor
Set up a cluster (both management and storage)	Y	Y	N	N	N	N	N



CHAPTER 19

Glossary

Terms	Description
BIOS	BIOS is firmware used to perform hardware initialization during the booting process, and to provide runtime services for operating systems and programs. The BIOS firmware comes pre-installed on a personal computer's system board, and it is the first software to run when powered on.
Cisco IMC	The Cisco Integrated Management Controller (IMC) is the management service for the C-Series servers. CIMC runs within the server. You can use a web-based GUI or SSH-based CLI to access, configure, administer, and monitor the server.
DPDK	The Data Plane Development Kit (DPDK) is a set of data plane libraries and network interface controller drivers for fast packet processing.
DTLS	Datagram Transport Layer Security (DTLS) is a communications protocol that provides security for datagram-based applications by allowing them to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery.
IPSec	Internet Protocol Security (IPsec) is a secure network protocol suite that authenticates and encrypts the packets of data to provide secure encrypted communication between two computers over an Internet Protocol network
LACP	Link Aggregation Control Protocol (LACP) is a protocol for the collective handling of multiple physical ports that can be seen as a single channel for network traffic purposes.
LLDP	Link Layer Discovery Protocol (LLDP) is a vendor independent link layer protocol used by network devices for advertising their identity, capabilities to neighbors on a LAN segment.
MIB	Management Information Base (MIB) is a database of the objects that can be managed on a device. The managed objects, or variables, can be set or read to provide information on the network devices and interfaces.
NETCONF	A Network Configuration Protocol (NETCONF) is a protocol defined by the IETF to install, edit, and delete the configuration of network devices.
NGIO	Next Generation Input/Output (NGIO)

Terms	Description
PnP	Plug and Play (PnP) increases speed and reduces complexity of device deployments.
Port Channels	Port channels combine individual links into a group to create a single logical link that provides the aggregate bandwidth of up to eight physical links.
RADIUS	Remote Authentication Dial-In User Service (RADIUS) is a networking protocol, operating on port 1812 that provides centralized Authentication, Authorization, and Accounting management for users who connect and use a network service.
REST API	Representational state transfer (REST) suggests to create an object of the data requested by the client and send the values of the object in response to the user.
Service Chaining	Service chaining allows data traffic to be rerouted through one or more services, such as firewall, load balancer, and intrusion detection and prevention (IDP) devices.
SNMP	Simple Network Management Protocol (SNMP) is a framework used for managing devices on the internet. It provides a set of operations for monitoring and managing the internet.
SPAN	Switched Port Analyzer (SPAN) feature, which is sometimes called port mirroring or port monitoring, selects network traffic for analysis by a network analyzer.
Spanning Tree	Spanning Tree Protocol (STP) is a Layer 2 protocol that runs on bridges and switches. The main purpose of STP is to ensure that you do not create loops when you have redundant paths in your network.
SR-IOV	Single Root I/O Virtualization (SR-IOV) allows multipleVMs running a variety of guest operating systems to share a single PCIe network adapter within a host server. SR-IOV allows a VM to move data directly to and from the network adapter, bypassing the hypervisor for increased network throughput and lower server CPU burden.
TACACS	Terminal Access Controller Access-Control System (TACACS) refers to a family of related protocols handling remote authentication and related services for networked access control through a centralized server.
UEFI	The Unified Extensible Firmware Interface (UEFI) is a specification that defines a software interface between an operating system and platform firmware.
virtio	Virtual input/output (virtio) is a virtualization standard for network and disk device drivers where just the guest's device driver "knows" it is running in a virtual environment, and cooperates with the hypervisor.
VM	A virtual machine (VM) is an emulation of a computer system. Virtual machines are based on computer architectures and provide functionality of a physical computer. Their implementations may involve specialized hardware, software, or a combination.
VNF	Virtual Network Functions (VNFs), the software version of network appliances such as a router, firewall, load-balancer etc