# NFVIS Monitoring



## Syslog

The Syslog feature allows event notifications from NFVIS to be sent to remote syslog servers for centralized log and event collection.The syslog messages are based on the occurrence of specific events on the device and provide configuration and operational information such as creation of users, changes to the interface status, and failed login attempts. Syslog data is critical to recording day-to-day events as well as notifying operational staff of critical system alerts.

Cisco enterprise NFVIS sends syslog messages to syslog servers configured by the user. Syslogs are sent for Network Configuration Protocol (NETCONF) notifications from NFVIS.

### Syslog Message Format

Syslog messages have the following format:

```
<Timestamp> hostname %SYS-<Severity>-<Event>: <Message>
```

Sample Syslog messages:

```
2017 Jun 16 11:20:22 nfvis %SYS-6-AAA_TYPE_CREATE: AAA authentication type tacacs created
successfully AAA authentication set to use tacacs server
2017 Jun 16 11:20:23 nfvis %SYS-6-RBAC_USER_CREATE: Created rbac user successfully: admin
2017 Jun 16 15:36:12 nfvis %SYS-6-CREATE_FLAVOR: Profile created: ISRv-small
2017 Jun 16 15:36:12 nfvis %SYS-6-CREATE_FLAVOR: Profile created: ISRv-medium
2017 Jun 16 15:36:13 nfvis %SYS-6-CREATE_IMAGE: Image created: ISRv_IMAGE_Test
2017 Jun 19 10:57:27 nfvis %SYS-6-NETWORK_CREATE: Network testnet created successfully
2017 Jun 21 13:55:57 nfvis %SYS-6-VM_ALIVE: VM is active: ROUTER
```

**Note** To refer to the complete list of syslog messages, see Syslog Messages

### 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 1: 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<br>• /api/operational/system/settings/logging | • system settings logging host<br>• system settings logging severity<br>• 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 2: 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<br>**Note**<br>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<br>**Note**<br>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<br>• /api/config/snmp/communities<br>• /api/config/snmp/enable/traps<br>• /api/config/snmp/hosts<br>• /api/config/snmp/user<br>• /api/config/snmp/groups | • agent<br>• community<br>• trap-type<br>• host<br>• user<br>• 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<br>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 3: 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.