



Cisco Nexus 1000V for Microsoft Hyper-V REST API Guide, Release 5.2(1)SM1(5.1)

June 10, 2013

Cisco Systems, Inc.

www.cisco.com

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

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.

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: www.cisco.com/go/trademarks. 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. (1110R)

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.

Cisco Nexus 1000V for Microsoft Hyper-V REST API Guide, Release 5.2(1)SM1(5.1)
© 2013 Cisco Systems, Inc. All rights reserved.



| | |
|--|----------|
| Preface | v |
| Audience | v |
| Recommended Reading | v |
| Document Organization | vi |
| Document Conventions | vi |
| Related Documentation | vii |
| Obtaining Documentation and Submitting a Service Request | viii |
| | viii |

CHAPTER 1

| | |
|---|------------|
| Overview | 1-1 |
| RESTful Web Services API | 1-1 |
| Finding Namespace Lists and Functions | 1-2 |
| List of Functions Available For N1K | 1-2 |
| List of Functions Available for Hyper-V | 1-3 |
| Create, Read, Update, and Delete Operations | 1-3 |
| Creating an Object | 1-3 |
| Reading an Object | 1-4 |
| Updating an Object | 1-4 |
| Deleting an Object | 1-5 |

CHAPTER 2

| | |
|--|------------|
| Hypervisor API Functions | 2-7 |
| Get License Information for the Cisco Nexus 1000V | 2-8 |
| Get Basic Information About the Cisco Nexus 1000V | 2-8 |
| Get Module Information for the Cisco Nexus 1000V | 2-9 |
| Get Uplink Information for the Cisco Nexus 1000V | 2-11 |
| Get Virtual Port Information for the Cisco Nexus 1000V | 2-12 |
| Get Port Profile Information for the Cisco Nexus 1000V | 2-13 |
| Get a List of Cisco Nexus 1000V Port Profiles on a Specific Module | 2-13 |
| Get a List of Cisco Nexus 1000V vNICs in Use on a Specific Module | 2-14 |
| Get a List of Cisco Nexus 1000V Uplink Ports In Use On a Specific Module | 2-15 |
| Get a List of Cisco Nexus 1000V VSEM Details | 2-15 |
| Get a List of Cisco Nexus 1000V Switch Extensions | 2-16 |

- [Get a List of Cisco Nexus 1000V Logical Networks](#) 2-18
- [Get a List of Cisco Nexus 1000V Network Segment Pools](#) 2-18
- [Get a List of Cisco Nexus 1000V IP Address Pools](#) 2-19
- [Get a List of Cisco Nexus 1000V Network Segments](#) 2-20
- [Get a List of Cisco Nexus 1000V VM Networks](#) 2-21
- [Get a List of Cisco Nexus 1000V Virtual Port Profiles](#) 2-22
- [Get a List of Cisco Nexus 1000V Uplink Port Profiles](#) 2-22



Preface

This preface introduces the Representational State Transfer (REST) application programming interface (API) functions for the Cisco Nexus 1000V for Microsoft Hyper-V and contains the following sections:

- [Audience, page v](#)
- [Recommended Reading, page v](#)
- [Document Organization, page vi](#)
- [Document Conventions, page vi](#)
- [Related Documentation, page vii](#)
- [Obtaining Documentation and Submitting a Service Request, page viii](#)

Audience

This guide is for network administrators with the following experience and knowledge:

- An understanding of virtualization
- Using tools to configure a virtual switch

Recommended Reading

Before configuring the Cisco Nexus 1000V for Microsoft Hyper-V, Cisco recommends that you read and become familiar with the following documentation:

- *Cisco Nexus 1000V for Microsoft Hyper-V Installation Guide, Release 5.2(1)SM1(5.1)*
- *Cisco Nexus 1000V for Microsoft Hyper-V Port Profile Configuration Guide, Release 5.2(1)SM1(5.1)*
- *Cisco Nexus 1000V for Microsoft Hyper-V NSM Configuration Guide, Release 5.2(1)SM1(5.1)*

Document Organization

This document is organized into the following chapters:

| Chapter | Description |
|---|---|
| Chapter 1, “Overview” | Provides a brief introduction on how to use the REST API. |
| Chapter 2, “Hypervisor API Functions” | Provides a detailed description of all of the REST API. |

Document Conventions

Command descriptions use these conventions:

| | |
|----------------------|---|
| boldface font | Commands and keywords are in boldface. |
| <i>italic font</i> | Arguments for which you supply values are in italics. |
| { } | Elements in braces are required choices. |
| [] | Elements in square brackets are optional. |
| x y z | Alternative, mutually exclusive elements are separated by vertical bars. |
| string | A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks. |

Screen examples use these conventions:

| | |
|-----------------------------|---|
| screen font | Terminal sessions and information the device displays are in screen font. |
| boldface screen font | Information you must enter is in boldface screen font. |
| <i>italic screen font</i> | Arguments for which you supply values are in italic screen font. |
| < > | Nonprinting characters, such as passwords, are in angle brackets. |
| [] | Default responses to system prompts are in square brackets. |
| !, # | An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line. |

This document uses the following additional conventions:



Note

Means reader *take note*. Notes contain helpful suggestions or references to material not covered in the manual.



Caution

Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Related Documentation

This section lists the documents used with the Cisco Nexus 1000V for Hyper-V.

General Information

Cisco Nexus 1000V for Hyper-V Release Notes

Install and Upgrade

Cisco Nexus 1000V for Hyper-V Installation Guide

Configuration Guides

Cisco Nexus 1000V for Hyper-V High Availability and Redundancy Configuration Guide

Cisco Nexus 1000V for Hyper-V Interface Configuration Guide

Cisco Nexus 1000V for Hyper-V Layer 2 Switching Configuration Guide

Cisco Nexus 1000V for Hyper-V License Configuration Guide

Cisco Nexus 1000V for Hyper-V Network Segmentation Manager Configuration Guide

Cisco Nexus 1000V for Hyper-V Port Profile Configuration Guide

Cisco Nexus 1000V for Hyper-V Quality of Service Configuration Guide

Cisco Nexus 1000V for Hyper-V Security Configuration Guide

Cisco Nexus 1000V for Hyper-V System Management Configuration Guide

Programming Guide

Cisco Nexus 1000V for Hyper-V REST API Guide

Reference and Troubleshooting Guides

Cisco Nexus 1000V for Hyper-V Command Reference

Cisco Nexus 1000V for Hyper-V Troubleshooting Guide

Virtual Services Appliance Documentation

The Cisco Nexus Virtual Services Appliance (VSA) documentation is available at http://www.cisco.com/en/US/products/ps9902/tsd_products_support_series_home.html

Virtual Security Gateway Documentation

The Cisco Virtual Security Gateway documentation is available at http://www.cisco.com/en/US/products/ps11208/tsd_products_support_model_home.html

Virtual Network Management Center

The Cisco Virtual Network Management Center documentation is available at http://www.cisco.com/en/US/products/ps11213/tsd_products_support_series_home.html

Virtual Wide Area Application Services (vWAAS)

The Virtual Wide Area Application Services documentation is available at
http://www.cisco.com/en/US/products/ps6870/tsd_products_support_series_home.html

ASA 1000V Cloud Firewall

The ASA 1000V Cloud Firewall documentation is available at
http://www.cisco.com/en/US/products/ps12233/tsd_products_support_series_home.html

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.



Overview

This chapter provides an overview for using the Representational State Transfer (REST) application programming interface (API) with the Cisco Nexus 1000V Series Switches.

This chapter contains the following sections:

- [RESTful Web Services API, page 1-1](#)
- [Finding Namespace Lists and Functions, page 1-2](#)
- [List of Functions Available For N1K, page 1-2](#)
- [List of Functions Available for Hyper-V, page 1-3](#)
- [Create, Read, Update, and Delete Operations, page 1-3](#)

RESTful Web Services API

The Cisco Nexus 1000V Virtual Supervisor Module (VSM) supports the RESTful webservices API and provides limited functionality through this service.

You can create, read, update, and delete an object on the Cisco Nexus 1000V VSM using the RESTful web services API. REST is based on HTTP and, therefore, these four operations are in turn mapped to GET, POST, and DELETE HTTP operations. In order to call any REST function, you can use tools such as a web browser, the cURL tool, and Windows PowerShell.

The following is the basic construct of a REST URL:

```
http[s]://<IP_address>/api/<object locator>
```

The object locator consists of two parts:

- <object locator> := <name space>/<object name>
- <name space> indicates the broader class of functions and <object name> refers to the specific object.

For example, in the following URL:

```
http://10.10.10.2/api/n1k/license
```

n1k is the namespace and license is the object name.

If you are using a browser, type in the URL. For example, if you want to get the license information of your VSM that has an IP address of 10.10.10.2, you type the URL as follows:

```
https://10.10.10.2/api/n1k/license
```

The browser prompts you for a username and a password. After entering them, you get the following output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<set name="license_set"
  <instance name="NEXUS_VSG_SERVICES_PKG" url="/api/n1k/license">
    <properties>
      <expires>Never</expires>
      <type>NEXUS_VSG_SERVICES_PKG</type>
      <available>16</available>
      <status>Unused</status>
      <used>0</used>
    </properties>
  </instance>
  <instance name="NEXUS_ASA1000V_SERVICES_PKG" url="/api/n1k/license">
    <properties>
      <expires>Never</expires>
      <type>NEXUS_ASA1000V_SERVICES_PKG</type>
      <available>16</available>
      <status>Unused</status>
      <used>0</used>
    </properties>
  </instance>
```

To access the same through cURL, you use the following format:

```
curl http://username:password@10.10.10.2/api/n1k/license
```


Note

When using the **reload** command through the REST tunnel, you need to make sure that the configuration is saved before using it. Otherwise the changes are lost.

Finding Namespace Lists and Functions

Every REST API function is associated with a namespace. Functions that are not specific to Hyper-V are under the n1k namespace. To find the list of namespaces, construct the URL as follows:

```
https://10.10.10.2/api
```

You get the following output:

```
<instance url="/api">
  <children>
    <child name="n1k" url="/api/n1k"/>
  </children>
</instance>
```

The above output tells you that there are two namespaces—`hyper-v` and `n1k`.

List of Functions Available For N1K

To find all the functions under the n1k namespace, you enter the following in the web browser:

```
https://10.10.10.2/api/n1k
```

You get the following output:

```
<instance url="/api/n1k">
```

```

<children>
  <child name="network-segment-pool" url="/api/nlk/network-segment-pool"/>
  <child name="uplink" url="/api/nlk/uplink"/>
  <child name="vnic" url="/api/nlk/vnic"/>
  <child name="network-segment" url="/api/nlk/network-segment"/>
  <child name="logical-network" url="/api/nlk/logical-network"/>
  <child name="ip-pool-template" url="/api/nlk/ip-pool-template"/>
  <child name="license" url="/api/nlk/license"/>
  <child name="summary" url="/api/nlk/summary"/>
  <child name="hyper-v" url="/api/nlk/hyper-v"/>
  <child name="port-profile" url="/api/nlk/port-profile"/>
  <child name="vem" url="/api/nlk/vem"/>
  <child name="uplink-port-profile" url="/api/nlk/uplink-port-profile"/>
  <child name="virtual-port-profile" url="/api/nlk/virtual-port-profile"/>
</children>
</instance>

```

Response Description

Each child shows the available functions under the current nlk namespace.

| Keyword | Description |
|---------|---|
| name | Name of a function. |
| uri | Relative uniform resource identifier (URI). |

List of Functions Available for Hyper-V

```

<instance url="/api/nlk/hyper-v">
  <children>
    <child name="vsem-system-info" url="/api/nlk/hyper-v/vsem-system-info"/>
    <child name="vm-network" url="/api/nlk/hyper-v/vm-network"/>
    <child name="switch-extension-info" url="/api/nlk/hyper-v/switch-extension-info"/>
  </children>
</instance>

```

Create, Read, Update, and Delete Operations

Creating an Object

To create an object, you must construct an HTTP POST request:

```
https://<IP address>/api/<name space>/<object locator>
```

The request must have a payload that contains JavaScript Object Notation- (JSON)-formatted fields that are part of the newly created object:

```
\{"<property>": "<value>", "<property>": "<value>", ....\}
```

For example, to create an IP address pool with a pool name of pool1 on a VSM with an IP address of 10.10.10.2, send a POST request by entering the following:

Use cURL to perform the following:

```
curl -u admin:Sfish123 10.10.10.2/api/nlk/ip-pool-template -d '{"name": "pool1", "addressRangeStart": "192.168.0.2", "addressRangeEnd": "192.168.0.16"}'
```

The same can be obtained through a browser using addons such as the REST console.

Use PowerShell to perform the following:

```
#Basic parameters required for accessing REST-APIs
$User = "admin"
$Password = ConvertTo-SecureString -String "Secret123" -AsPlainText -Force
$VSMIPAddress = "10.10.10.2"
$URI = "http://" + $VSMIPAddress
$Credential = New-Object -TypeName System.Management.Automation.PSCredential
-ArgumentList $User, $Password
#Create-
$args = ' {"name" : "pool1" , "addressRangeStart": "192.168.0.2" ,
"addressRangeEnd": "192.168.0.16"}'
ConvertFrom-Json -InputObject $args
Invoke-RestMethod -Uri http://10.10.10.2/api/n1k/ip-pool-template -Credential
$Credential -Method Post -Body $args
```

To find out the valid property names for a given function, see the Response Sample in each function definition later in this document.

Reading an Object

To read an object, you must construct an HTTP GET request:

```
https://<IP address>/api/<name space>/<object locator>/<instance name>
```

For example, to read switch extension manager information from a VSM with an IP address of 10.10.10.2, send a GET request by entering the following:

Use cURL to perform the following:

```
curl -u admin:Sfish123 10.10.10.2/api/n1k/hyper-v/vsem-system-info
```

Use PowerShell to perform the following:

```
#Read the VSEM info - HTTP GET
$VersionURI = $URI + "/api/hyper-v/vsem-system-info"
Invoke-RestMethod -Uri $VersionURI -Credential $Credential -Method Get -Outfile
testout.xml
```

Sends the results to the specified output file. In this case, "testout.xml" . Enter \ a path and file name. If you omit the path, the default is the current location.

Updating an Object

To update an object, you must construct an HTTP POST request:

```
https://<IP address>/api/<name space>/<object locator>/<instance name>
```

The request must have a payload that contains JSON-formatted fields that are updated on the object:

```
\{"<property>": "<value>", "<property>": "<value>", .....}
```

For example, to modify the address range for the IP pool named pool1 on a VSM with an IP address of 10.10.10.2, send a POST request by entering the following IP pool:

Use cURL to perform the following:

```
curl -u admin:Sfish123 10.10.10.2/api/n1k/ip-pool-template/pool1 -d '{"name":"pool1",
"addressRangeStart":"192.168.0.5", "addressRangeEnd":"192.168.0.20"}'
```

Use PowerShell to perform the following:

```
#Update the IP-address Pool Information - HTTP POST
$IPPURI = $URI + "/api/n1k/ip-pool-template/pool1"
$IPPArg = '{"addressRangeStart":"192.168.0.5", "addressRangeEnd":"192.168.0.20"}'
ConvertFrom-Json -InputObject $IPPArg
Invoke-RestMethod -Uri $IPPURI -Credential $Credential -Method Post -Body $IPPArg
```

To find the valid property names for a given function, see the Response Sample in each function definition later in this document.

Deleting an Object

To delete an object, you must construct an HTTP DELETE request:

```
https://<IP address>/api/<name space>/<object locator>/<instance name>
```

To delete a network segment named VMN4 from a VSM with an IP address of 10.10.10.2, send a DELETE request by entering the following:

Use cURL to perform the following:

```
curl -u admin:Sfish123 10.10.10.2/api/n1k/network-segment/VMN4 -X DELETE
```

Use PowerShell to perform the following:

```
#Delete a network segment - HTTP Delete
$VMNURI = $URI + "/api/n1k/network-segment/VMN4"
Invoke-RestMethod -Uri $VMNURI -Credential $Credential -Method Delete
```




Hypervisor API Functions

This chapter provides information about the Cisco Nexus 1000V application programming interface (API) functions. The functions are grouped under two namespaces—`n1k` and `hyper-v`. The `n1k` namespace is generic across all hypervisors, and `hyper-v` namespaces are specific. The functions that support write (create, update, and delete) operations are explicitly marked as writeable. All others are read-only functions.

The functions are the following:

- [Get License Information for the Cisco Nexus 1000V](#), page 2-8
- [Get Basic Information About the Cisco Nexus 1000V](#), page 2-8
- [Get Module Information for the Cisco Nexus 1000V](#), page 2-9
- [Get Uplink Information for the Cisco Nexus 1000V](#), page 2-11
- [Get Virtual Port Information for the Cisco Nexus 1000V](#), page 2-12
- [Get Port Profile Information for the Cisco Nexus 1000V](#), page 2-13
- [Get a List of Cisco Nexus 1000V Port Profiles on a Specific Module](#), page 2-13
- [Get a List of Cisco Nexus 1000V vNICs in Use on a Specific Module](#), page 2-14
- [Get a List of Cisco Nexus 1000V Uplink Ports In Use On a Specific Module](#), page 2-15
- [Get a List of Cisco Nexus 1000V VSEM Details](#), page 2-15
- [Get a List of Cisco Nexus 1000V Switch Extensions](#), page 2-16
- [Get a List of Cisco Nexus 1000V Logical Networks](#), page 2-18
- [Get a List of Cisco Nexus 1000V Network Segment Pools](#), page 2-18
- [Get a List of Cisco Nexus 1000V IP Address Pools](#), page 2-19
- [Get a List of Cisco Nexus 1000V Network Segments](#), page 2-20
- [Get a List of Cisco Nexus 1000V VM Networks](#), page 2-21
- [Get a List of Cisco Nexus 1000V Virtual Port Profiles](#), page 2-22
- [Get a List of Cisco Nexus 1000V Uplink Port Profiles](#), page 2-22

Get License Information for the Cisco Nexus 1000V

Object Locator

[“/api/n1k/license”]

Description

Retrieves license-usage information. This function is equivalent to the **show license usage** command.

Response Sample

```
<?xml version="1.0" encoding="UTF-8"?>
<set name="license_set"
  <instance name="NEXUS_VSG_SERVICES_PKG" url="/api/n1k/license">
    <properties>
      <expires>Never</expires>
      <type>NEXUS_VSG_SERVICES_PKG</type>
      <available>16</available>
      <status>Unused</status>
      <used>0</used>
    </properties>
  </instance>
  <instance name="NEXUS_ASA1000V_SERVICES_PKG" url="/api/n1k/license">
    <properties>
      <expires>Never</expires>
      <type>NEXUS_ASA1000V_SERVICES_PKG</type>
      <available>16</available>
      <status>Unused</status>
      <used>0</used>
    </properties>
  </instance>
  <instance name="N1KV_MSFT_LAN_SERVICES_PKG" url="/api/n1k/license">
    <properties>
      <expires>Never</expires>
      <type>N1KV_MSFT_LAN_SERVICES_PKG</type>
      <available>511</available>
      <status>In use</status>
      <used>0</used>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description |
|-----------|---|
| expires | Earliest expiration date for a given license. |
| type | License name. |
| available | Available license count. |
| status | License usage status. |
| used | Used licenses. |

Get Basic Information About the Cisco Nexus 1000V

Object Locator

[“/api/n1k/summary”]

Description

Retrieves basic information about the Cisco Nexus 1000V.

Response Sample

```

<instance uri="/api/n1k/summary">
  <properties>
    <switchMode>Advance</switchMode>
    <ip>172.23.231.209</ip>
    <name>my_vsm</name>
    <haStatus>>false</haStatus>
    <version>version 5.2(1)SM1(5.1)</version>
  </properties>
</instance>

```

Response Description

| Property | Description |
|------------|---|
| switchMode | Switch mode. Can be Essential or Advanced. |
| ip | IP address of the Cisco Nexus 1000V VSM. |
| name | Cisco Nexus 1000V VSM hostname. |
| haStatus | Whether the VSM is in high availability mode. |
| version | Cisco Nexus 1000V version. |

Get Module Information for the Cisco Nexus 1000V

Object Locator

["/api/n1k/vem"]

Description

Retrieves information about Cisco Nexus 1000V modules. This function is equivalent to the **show module** command.

Response Sample

```

<set name="vem_set">
  <instance name="3" url="/api/n1k/vem">
    <properties>
      <module>3</module>
      <licenseUsage>-</licenseUsage>
      <numVnics>3</numVnics>
      <ip>10.10.10.2</ip>
      <hostVersion>Windows Server 8 - Datacenter (6.2.9200, 6.30)
      </hostVersion>
      <status>ok</status>
      <license>licensed</license>
      <mac>02-00-0c-00-03-00 to 02-00-0c-00-03-80</mac>
      <type>Virtual Ethernet Module</type>
      <maxNumVnic>216</maxNumVnic>
      <nSockets>2</nSockets>
      <ports>288</ports>
      <modmac>3</modmac>
    </properties>
  </instance>
</set>

```

```

    <numVM>3</numVM>
    <version>5.2(1)SM1(5.1)</version>
    <model>NA</model>
    <lic_version>-</lic_version>
    <serialnum>NA</serialnum>
    <name>SAMLIN-SERVER02</name>
  </properties>
  <children>
    <child name="port-profile" url="/api/n1k/vem/3/port-profile"/>
    <child name="uplink" url="/api/n1k/vem/3/uplink"/>
    <child name="vnic" url="/api/n1k/vem/3/vnic"/>
  </children>
</instance>
</set>

```

Response Description

| Property | Description |
|--------------|---|
| module | Module number. |
| licenseUsage | Number of CPU licenses used by the Virtual Ethernet Module (VEM). |
| numVnics | Number of virtual network interface cards (vNICs) on the host. |
| ip | IP address of the host. |
| hostVersion | Host version. |
| status | Host status. |
| license | License status. |
| mac | Host MAC address. |
| type | Host type. |
| nSockets | Number of sockets on the host. |
| ports | Number of available ports per module. |
| modmac | Number of MAC addresses learned on the module. |
| numVM | Number of active virtual machines (VMs) on the host. |
| model | Host model. |
| lic_version | Licensing version. |
| serialnum | Serial number of the host. |
| name | Host DNS name. |

Each child indicates the sub-addon available under each module.

Get Uplink Information for the Cisco Nexus 1000V

Object Locator

["/api/n1k/uplink"]

Description

Retrieves information about the Cisco Nexus 1000V uplink ports.

Response Sample

```
<set name="uplink_set">
  <instance name="Ethernet3/8" uri="/api/n1k/uplink/Ethernet3%2F8">
    <properties>
      <module>3</module>
      <portChannelType>Eth</portChannelType>
      <packetsTx>26361</packetsTx>
      <mtu>1500</mtu>
      <cdpPort>GigabitEthernet3/39</cdpPort>
      <port>Ethernet3/8</port>
      <status>up</status>
      <mode>trunk</mode>
      <vlans>231</vlans>
      <portChannel>1</portChannel>
      <cdpNativeVlan>231</cdpNativeVlan>
      <portChannelMembers>Ethernet3/8</portChannelMembers>
      <ethernet>Ethernet</ethernet>
      <packetsRx>187544</packetsRx>
      <cdpSwitch>sfish-6k-I9</cdpSwitch>
      <portProfile>n1kv-uplink0</portProfile>
      <speed>1000 Mb/s</speed>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description |
|--------------------|--|
| module | Module ID to which the uplink is connected. |
| portChannelType | Port channel type. |
| packetsTx | Number of packets transmitted. |
| mtu | Maximum transmission unit (MTU) size. |
| cdpPort | Name of the port of the Cisco Discovery Protocol neighbor. |
| port | Uplink port name. |
| status | Uplink status. |
| mode | Uplink port mode (access or trunk). |
| vlans | VLAN associated with the uplink. |
| portChannel | Port channel group. |
| cdpNativeVlan | Native VLAN of the Cisco Discovery Protocol neighbor. |
| portChannelMembers | Members of the port channel. |
| ethernet | Type of port. |

| Property | Description |
|-------------|---|
| packetsRx | Number of packets sent. |
| cdpSwitch | Name of the Cisco Discovery Protocol neighbor switch. |
| portProfile | Name of the assigned port profile. |
| speed | Uplink speed. |

Get Virtual Port Information for the Cisco Nexus 1000V

Object Locator

[/api/n1k/vnic]

Description

Retrieves information about the Cisco Nexus 1000V virtual ports.

Response Sample

```
<set name="vnic_set">
  <instance name="Vethernet5" uri="/api/n1k/vnic/Vethernet5">
    <properties>
      <mac>0050.56ba.58bc</mac>
      <dvport>DVPort65</dvport>
      <adapter>Net Adapter 1</adapter>
      <hostIP>172.23.231.192</hostIP>
      <vlans>231</vlans>
      <portGroup>n1kv-system-control</portGroup>
      <status>up</status>
      <module>3</module>
      <vm>vsm-openstack</vm>
      <vnic>Vethernet5</vnic>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description |
|-----------|---|
| mac | MAC address associated with the vNIC. |
| dvport | Distributed virtual port Globally Unique Identifier (GUID). |
| adapter | Adapter with which the vNIC is associated. |
| hostIP | IP address of the host vNIC. |
| vlans | VLANs assigned to the vNIC. |
| portGroup | Port profile to which the vNIC is assigned. |
| status | vNIC status. |
| module | Module ID with which the vNIC is associated. |
| vm | Virtual machine to which the vNIC is connected. |
| vnic | vNIC name. |

Get Port Profile Information for the Cisco Nexus 1000V

Object Locator

[/api/n1k/port-profile]

Description

Retrieves information about the Cisco Nexus 1000V port profiles.

Response Sample

```
<set name="port-profile_set">
  <instance name="n1kv-pp-vmk0" uri="/api/n1k/port-profile/n1kv-pp-vmk0">
    <properties>
      <minPorts>1</minPorts>
      <systemVlans>231</systemVlans>
      <usedPorts>1</usedPorts>
      <name>n1kv-pp-vmk0</name>
      <vlans>231</vlans>
      <status>1</status>
      <mode>access</mode>
      <maxPorts>32</maxPorts>
      <type>Vethernet</type>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description |
|-------------|---|
| minPorts | Minimum number of ports allowed per port profile. |
| systemVlans | Assigned system VLANs. |
| usedPorts | Used ports. |
| name | Name of the port profile. |
| vlans | Number of associated VLANs. |
| status | Status of the port profile. |
| mode | Mode of the port profile (access, trunk). |
| maxPorts | Maximum number of ports allowed per port profile. |
| type | Type of port profile (vEthernet, Ethernet). |

Get a List of Cisco Nexus 1000V Port Profiles on a Specific Module

Object Locator

[/api/n1k/vem/<module number>/port-profile]

Description

Retrieves a list of the Cisco Nexus 1000V port profiles in use on a specific module.

Response Sample

```
<set name="port-profile_set">
  <instance name="1" uri="/api/n1k/vem/3/port-profile/1">
    <properties>
      <minPorts>1</minPorts>
      <systemVlans>231</systemVlans>
      <usedPorts>1</usedPorts>
      <name>n1kv-pp-vmk0</name>
      <vlans>231</vlans>
      <status>1</status>
      <mode>access</mode>
      <maxPorts>32</maxPorts>
      <type>Vethernet</type>
    </properties>
  </instance>
</set>
```

Response Description

See the description for [/api/n1k/port-profile].

Get a List of Cisco Nexus 1000V vNICs in Use on a Specific Module

Object Locator

["/api/n1k/vem/<module no>/vnic/<vnic number>"]

Description

Retrieves a list of Cisco Nexus 1000V virtual ports in use on a specific module.

Response Sample

```
<set name="vnic_set">
  <instance name="Vethernet5" uri="/api/n1k/vem/3/vnic/Vethernet5">
    <properties>
      <mac>0050.56ba.58bc</mac>
      <dvport>DVPort65</dvport>
      <adapter>Net Adapter 1</adapter>
      <hostIP>172.23.231.192</hostIP>
      <vlans>231</vlans>
      <portGroup>n1kv-system-control</portGroup>
      <status>up</status>
      <module>3</module>
      <vm></module>
      <vnic>Vethernet5</vnic>
    </properties>
  </instance>
</set>
```

Response Description

See the description for [/api/n1k/vnic].

Get a List of Cisco Nexus 1000V Uplink Ports In Use On a Specific Module

Object Locator

["/api/n1k/vem/<module number>/uplink"]

Description

Retrieves a list of uplink ports in use on a specific module and related information.

Response Sample

```
<set name="uplink_set">
  <instance name="Ethernet3/8" uri="/api/n1k/vem/3/uplink/Ethernet3%2F8">
    <properties>
      <module>3</module>
      <portChannelType>Eth</portChannelType>
      <packetsTx>33077</packetsTx>
      <mtu>1500</mtu>
      <cdpPort>GigabitEthernet3/39</cdpPort>
      <port>Ethernet3/8</port>
      <status>up</status>
      <mode>trunk</mode>
      <vlans>231</vlans>
      <portChannel>1</portChannel>
      <cdpNativeVlan>231</cdpNativeVlan>
      <portChannelMembers>Ethernet3/8</portChannelMembers>
      <ethernet>Ethernet</ethernet>
      <packetsRx>235849</packetsRx>
      <cdpSwitch>sfish-6k-I9</cdpSwitch>
      <portProfile>n1kv-uplink0</portProfile>
      <speed>1000 Mb/s</speed>
    </properties>
  </instance>
</set>
```

Response Description

See the description for [/api/n1k/uplink].

Get a List of Cisco Nexus 1000V VSEM Details

Object Locator

["/api/n1k/hyper-v/vsem-system-info"]

Description

Retrieves information that is associated with the Virtual Switch Extension Module (VSEM) object. For any external switch device managed by the Virtual Machine Manager (VMM), the VMM associates it to a VSEM object.

Response Sample

```
<instance name="vsem-system-info" uri="/api/n1k/hyper-v/vsem-system-info">
  <properties>
    <description>Cisco Systems Nexus 1000V</description>
    <model>Nexus 1000V</model><id></model>
    <manufacturer>Cisco Systems</manufacturer>
    <name>Nexus 1000V Chassis version 5.2(1)SM1(5.1) [build
      5.2(1)SM1(5.0.128)|build 5.2(1)SM1(5.0.128)]</name>
    <version>1.0</version>
    <id>277cebaa-5c64-1336-a9d5-60965f468a4c</id>
    <vendorId>{55ca4f11-f549-4440-a489-e7337f3a6b73}</vendorId>
  </properties>
</instance>
```

Response Description

| Property | Description |
|--------------|---|
| description | VSEM description. |
| model | Model of the VSEM. |
| manufacturer | Manufacturer name. |
| name | User-readable name for a VSEM instance. |
| version | VSEM version. |
| id | VSEM ID. |
| vendorId | Vendor ID. |

Get a List of Cisco Nexus 1000V Switch Extensions

Object Locator

["/api/n1k/hyper-v/switch-extension-info"]

Description

Retrieves the VSM information needed by VMM to associate it to a switch extension. A VSEM can contain one or more switch extensions. For the Cisco Nexus 1000V, a VSEM can contain only one switch extension.

Response Sample

```
<instance url="/api/n1k/hyper-v/switch-extension-info">
  <properties>
    <drivernetcfginstanceid>9C8ED422-F33A-4F34-B771-E8B8D0539FD3
    </drivernetcfginstanceid>
    <name>hyperv-vsm-1</name>
    <opdata>data-version</opdata>
    <maxVersion>5.2.128</maxVersion>
    <minVersion>105.100.0000.0000</minVersion>
    <extensionType>Forwarding</extensionType>
    <isSwitchTeamSupported>true</isSwitchTeamSupported>
    <switchExtensionFeatureConfigId>2ABD62F9-0E77-4E4C-B7B0-B2DBAF9B7CBB
    </switchExtensionFeatureConfigId>>
    <maxNumberOfPorts>16000</maxNumberOfPorts>
    <mandatoryFeatureId>2ABD62F9-0E77-4E4C-B7B0-B2DBAF9B7CBB
    </mandatoryFeatureId>
    <maxNumberOfPortsPerHost>216</maxNumberOfPortsPerHost>
    <isChildOfWFPSwitchExtension>false</isChildOfWFPSwitchExtension>
  </properties>&nbsp;
```



```
</instance>
```

Response Description

| Property | Description |
|-----------------------------|---|
| drivernetcfginstanceid | Unique ID of the switch extension driver. |
| name | User-friendly name for the switch extension information. |
| opdata | Opaque data. |
| id | Unique ID of the switch extension information. |
| maxVersion | Maximum version of the switch extension driver that is supported. |
| minVersion | Minimum version of the switch extension driver that is supported. |
| extensionType | Extension type of the Nexus 1000V switch extension driver. The Nexus 1000V is a Forwarding extension. |
| isSwitchTeamSupported | Whether the switch supports Teaming. For the Cisco Nexus 1000V, the answer is Yes. |
| maxNumberOfPorts | Maximum number of ports that can be created. |
| mandatoryFeatureId | Mandatory feature ID. This is not set for a Cisco Nexus 1000V. |
| maxNumberOfPortsPerHost | Maximum number of ports per host that can be created on a logical switch. |
| isChildOfWFPSwitchExtension | Not set for a Cisco Nexus 1000V. |

Get a List of Cisco Nexus 1000V Logical Networks

Object Locator

[“/api/n1k/logical-network”] [writeable]

Description

Retrieves a list of logical networks. This function represents a logical network that spans across multiple sites represented by multiple subnets.

Response Sample

```
<set name="logical_network_set">
  <instance name="LN1" url="/api/n1k/logical-network">
    <properties>
      <name>LN1</name>
      <description>LN1</description>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description | Writeable? |
|-------------|--|------------|
| name | User field name for a logical network. | No |
| description | Description of a logical network. | Yes |

Get a List of Cisco Nexus 1000V Network Segment Pools

Object Locator

[“/api/n1k/network-segment-pool”] [writeable]

Description

Retrieves a list of network segment pools. The network segment pools contain one or more network segments that represent a logical network in a location. This function is equivalent to the VMM network segment pool.

Response Sample

```
<set name="network_segment_pool_set">
  <instance name="my-network-segment-pool" url="/api/n1k/network-segment-pool">
    <properties>
      <name>my-network-segment-pool</name>
      <logicalNetwork>my-logical-network</logicalNetwork>
      <id>a3e7d0b7-98ac-44f7-89ad-0eaa9675d753</id>
      <maximumNetworkSegmentsPerVMNetwork>2000</maximumNetworkSegmentsPerVMNetwork>
      <supportsVMNetworkProvisioning>true</supportsVMNetworkProvisioning>
      <supportsIpPool>true</supportsIpPool>
      <intraPortCommunication>true</intraPortCommunication>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description | Writeable? |
|---|--|------------|
| name | Username for the fabric network. | No |
| logicalNetwork | Fabric network name. | Yes |
| id | Unique ID of the network segment pool. | Yes |
| maximumNetworkSegmentsPerVMNetwork | Maximum number of network segments that can belong to a network segment. | Yes |
| supportsVMNetworkProvisioning | Whether network segment provisioning is supported. | Yes |
| supportsIpPool | Network segments in this pool that can have an associated IP pool. | Yes |
| maximumVMNetworkDefinitionsPerVMNetwork | Maximum network segment definitions per network segment. | Yes |
| intraPortCommunication | Whether intraport communication exists. | Yes |

Get a List of Cisco Nexus 1000V IP Address Pools

Object Locator

[“/api/n1k/ip-pool-template”] [writeable]

Description

Retrieves the Layer-3 information associated with the Layer-2 network. Every network segment is associated with an IP address pool.

Response Sample

```
<set name="ip_address_pool_set">
  <instance name="ip-pool-template" url="api/n1k/ip-pooltemplate">
    <properties>
      <netbt>>false</netbt>
      <description/>
      <addressRangeStart>1.1.1.2</addressRangeStart>
      <dhcp>>true</dhcp>
      <dnsSuffixList>cisco.com</dnsSuffixList>
      <addressFamily>IPv4</addressFamily>
      <dnsServersListList>1.1.1.3</dnsServersListList>
      <ipAddressSubnet>255.255.255.0</ipAddressSubnet>
      <addressRangeEnd>1.1.1.100</addressRangeEnd>
      <networkAddress>1.1.1.1</networkAddress>
      <name>ip-pool-template</name>
      <gateway>1.1.1.1</gateway>
      <netbiosServersList>1.1.1.99</netbiosServersList>
      <reservedIpList>1.1.1.50</reservedIpList>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description | Writeable? |
|-------------------|--|------------|
| description | Description of an IP address pool. | Yes |
| dhcp | Whether the network is DHCP-supported. | Yes |
| dnsSuffixList | DNS suffix. | Yes |
| addressRangeStart | Start of the IP address range. | Yes |
| addressRangeEnd | End of the IP address range. | Yes |
| ipAddressSubnet | IP address subnet. | Yes |
| winServerList | List of IP addresses of WINServer system in order of use. | Yes |
| name | IP pool name. | No |
| gateway | Network gateways. | Yes |
| netbios | Whether NetBIOS is enabled. | Yes |
| reservedIpList | List of static IP addresses excluded from allocation by VMM. | Yes |
| dnsServerList | IP addresses of DNS servers in order of use. | Yes |
| addressFamily | Family IP address. | Yes |

Get a List of Cisco Nexus 1000V Network Segments

Object Locator

["/api/n1k/network-segment"] [writeable]

Description

Retrieves a list of network segments. A network segment represents a subnet.

```
<set name="network_segment_set">
  <instance name="my-network-segment-secondary-101" url="/api/n1k/network-segment">
    <properties>
      <description>My secondary (101) network segment</description>
      <ipPoolId>d4fe3b3a-8ade-4fd0-9e3d-0af43b760c1a</ipPoolId>
      <id>21e1dbbe-7fb4-456c-a91b-8c66ff33792b</id>
      <vmNetworkId>5b602e01-58c8-4b48-a3b8-9d66c5dac5d9</vmNetworkId>
      <segmentType>VLAN</segmentType>
      <networkSegmentPool>my-network-segment-pool</networkSegmentPool>
      <name>joe-network-segment-secondary-101</name>
      <vmNetwork>my-network-segment-secondary-101</vmNetwork>
      <vlan>0</vlan>
      <ipPoolName>joe-ip-pool-template</ipPoolName>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description | Writeable? |
|--------------------|--|------------|
| description | Name given to network segment pool. | No |
| ipPoolId | Unique ID of the IP pool instance. | Yes |
| id | Unique ID of the network segment. | Yes |
| vmNetworkId | Unique ID of network segment. | Yes |
| segmentType | Type of segment. | Yes |
| networkSegmentPool | Network segment pool that this network segment is a member of. | Yes |
| name | Name of network segment. | Yes |
| vmNetwork | Network segment object name. | Yes |
| vlan | Associated VLAN. | Yes |
| ipPoolName | Available IP pool template name. | Yes |

Get a List of Cisco Nexus 1000V VM Networks

Object Locator

```
["/api/n1k/hyper-v/vm-network"] [writeable]
```

Description

Retrieves a list of virtual machine (VM) networks. A network segment can contain one or more network segments. For VLAN-based networks, a network segment can contain only one network segment.

Response Sample

```
<set name="vm_network_set">
  <instance name="my-network-segment-access-110" url="/api/n1k/hyper-v/vm-network">
    <properties>
      <id>26735817-8b5f-4a84-94e8-da72aee20ad5</id>
      <networkSegment>my-network-segment-access-110</networkSegment>
      <name>joe-network-segment-access-110</name>
      <networkSegmentPool>my-network-segment-pool</networkSegmentPool>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description | Writeable? |
|--------------------|--|------------|
| id | Unique ID of the network segment. | Yes |
| networkSegment | User-friendly name of the network segment. | Yes |
| name | User-friendly name of the network segment. | No |
| networkSegmentPool | Network segment pool. | Yes |

Get a List of Cisco Nexus 1000V Virtual Port Profiles

Object Locator

[“/api/n1k/virtual-port-profile”]

Description

Retrieves a list of virtual port profiles. A virtual port profile is a port profile that can be attached to a virtual interface.

Response Sample

```
<set name="virtual_port_profile_set">
  <instance name="veth-pp" url="/api/n1k/virtual-port-profile">
    <properties>
      <state>enabled</state>
      <type>vethernet</type>
      <name>veth-pp</name>
      <id>4b87bc57-e686-4296-8720-92736aec95b</id>
      <maxNumberOfPortsPerHost>216</maxNumberOfPortsPerHost>
      <maxPorts>32</maxPorts>
      <switchId>86e13b66-ba94-4190-9e5d-e2e43c9ec1cd</switchId>
    </properties>
  </instance>
</set>
```

Response Description

| Property | Description |
|-------------------------|---|
| state | State of the port profile. |
| type | Type of port profile. |
| name | User-friendly profile name. |
| id | Unique profile ID. |
| maxNumberOfPortsPerHost | Maximum number of ports per host. |
| maxPorts | Port capacity. |
| switchId | ID of the switch extension to which to publish this port profile. |

Get a List of Cisco Nexus 1000V Uplink Port Profiles

Object Locator

[“/api/n1k/uplink-port-profile”]

Description

Retrieves a list of uplink port profiles. An uplink port profile is a port profile that can be attached to a physical interface.

Response Sample

```
<set name="uplink-port-profile_set">
  <instance name="UPP1" url="/api/n1k/uplink-port-profile">
    <properties>
      <switchId>86e13b66-ba94-4190-9e5d-e2e43c9ec1cd</switchId>
```

```
<name>UPP1</name>
<id>7978f13c-70ad-4ecd-peec-51c8027ef629</id>
<maxPorts>32</maxPorts>
<networkSegmentPool>nsp1</networkSegmentPool>
</properties>
</instance>
</set>
```

Response Description

| Property | Description |
|--------------------|---|
| switchId | ID of the switch extension on which to publish this port profile. |
| name | User-friendly profile name. |
| id | Unique profile ID. |
| maxPorts | Port capacity. |
| networkSegmentPool | Network segment pool. |

