Cisco Nexus 1000V Troubleshooting Guide, Release 5.2(1)SV3(1.1)

July 9, 2020

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.

Text Part Number: OL-31393-01

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 used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

CONTENTS

New and Changed Information

This section describes the information in this document that is either new or has changed with each release.

To find additional information about new features or command changes, see the following:

- Release Notes.
- Command Reference.

Feature	Description	Changed in release	Where Documented
VSI Discovery and Configuration Protocol	Added new section for troubleshooting commands for the VSI Discovery and Configuration Protocol (VDP).	4.2(1)SV2(2.2)	VSI Discovery and Configuration Protocol
VXLAN Gateway	Added a section for troubleshooting commands for VXLAN Gateway. **Note** Starting with Release 5.2(1)SV3(1.15), Cisco Nexus 1000V for VMware vSphere does not support the VXLAN Gateway feature.	4.2(1)SV2(2.1)	VXLANs
Upgrade	Added section for problems with VSM-VEM Layer 2 to 3 Conversion Tool.	4.2(1)SV2(1.1)	Upgrades
Ethanalyzer	Added Ethanalyzer as a Cisco Nexus 1000V protocol analyzer tool content.	4.2(1)SV2(1.1)	Ethanalyzer
DHCP Enhancements	Added the troubleshooting commands for DHCP.	4.2(1)SV2(1.1)	DHCP, DAI, and IPSG Troubleshooting Commands
High Availability	Updated the high availability section. Added a command output for the new **show system internal active-active remote accounting logs** command and updated the output for the **show system redundancy status** command.	4.2(1)SV2(1.1)	High Availability

Feature	Description	Changed in release	Where Documented
Licensing	Added the **svs license transfer src-vem** *vem no* **license_pool** command to troubleshoot the issues with checking out the licenses or returning them to the license pool.	4.2(1)SV2(1.1)	License Troubleshooting Commands
Nexus 1000V VC Plugin Installation	Added a new section to troubleshoot the Cisco Nexus 1000V VC plugin installation.	4.2(1)SV2(1.1)	vCenter Plug-in
Nexus 1000V Installation Management Center	Added a new section to troubleshoot the Cisco Nexus 1000V Installation Management Center.	4.2(1)SV1(5.1)	Problems with the Cisco Nexus 1000V Installation Management Center
Recovering Management and Control Connectivity of a Host	Added a new section to recover management and control connectivity of a host when a VSM is running on a VEM.	4.2(1)SV1(5.1)	Recovering Management and Control Connectivity of a Host When a VSM is Running on a VEM
ACL Logging	Added a new section to troubleshoot ACL Logging.	4.2(1)SV1(5.1)	Troubleshooting ACL Logging
NSM	Added a new chapter to troubleshoot the Network Segmentation Manager (NSM).	4.2(1)SV1(5.1)	Network Segmentation Manager
VXLAN	Added a new chapter to troubleshoot the Virtual Extensible Local Area Network (VXLAN).	4.2(1)SV1(5.1)	VXLANs
Microsoft NLBUnicast Mode	Added a new section for troubleshooting Microsoft Network Load Balancing (NLB) unicast mode.	4.2(1)SV1(5.1)	Layer 2 Switching
In service software upgrade (ISSU)	Added a new section for troubleshooting ISSU.	4.2(1)SV1(4a)	Upgrades
VEM software upgrade	Added a new section for troubleshooting a VEM software upgrade.	4.2(1)SV1(4a)	Upgrades
DHCP, DAI, IPSG	Added a new section for troubleshooting DHCP, Dynamic ARP Inspection, and IP Source Guard.	4.2(1)SV1(4)	DHCP, DAI, and IPSG
Port profiles	Added a new section for port profiles and new information about quarantined port profiles.	4.2(1)SV1(4)	Port Profiles
Upgrade	Added a new section for troubleshooting upgrade problems.	4.2(1)SV1(4)	Upgrades
VEM health check	Added information about the VEM health check that shows the cause of a connectivity problem.	4.0(4)SV1(3)	Checking Network Connectivity Between the VSM and the VEM

Feature	Description	Changed in release	Where Documented
Storm Control	Added information about how to identify and resolve the problems related to storm control.	5.2(1)SV3(1.1)	Storm Control
L3Sec	Added information about how to secure the internal control plane communications (Control and Packet traffic) of Cisco Nexus 1000V in a more robust way than in previous releases. It operates only in Layer 3 control mode.	5.2(1)SV3(1.1)	L3Sec

Preface

The Troubleshooting document provides information about how to recognize a problem, determine its cause, and find possible solutions.

This preface describes the following aspects of this document:

Audience

This publication is for experienced network administrators who configure and maintain a Cisco Nexus 1000V.

Document Conventions

Command descriptions use these conventions:

Convention	Description
boldface font	Commands and keywords are in boldface.
italic font	Arguments for which you supply values are in italics.
[ ]	Elements in square brackets are optional.
[x | y | z]	Optional alternative keywords are grouped in brackets and 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 that the switch displays are in screen font.
`boldface screen font`	Information that you must enter is in boldface screen font.

`italic screen font`	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 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 1000 and available on Cisco.com at the following URL:

http://www.cisco.com/en/US/products/ps9902/tsd_products_support_series_home.html

General Information

Cisco Nexus 1000V Documentation Roadmap

Cisco Nexus 1000V Release Notes

Cisco Nexus 1000V Compatibility Information

Install and Upgrade

Cisco Nexus 1000V Installation and Upgrade Guide

Configuration Guides

Cisco Nexus 1000V High Availability and Redundancy Configuration Guide

Cisco Nexus 1000V Interface Configuration Guide

Cisco Nexus 1000V Layer 2 Switching Configuration Guide

Cisco Nexus 1000V License Configuration Guide

Cisco Nexus 1000V Network Segmentation Manager Configuration Guide

Cisco Nexus 1000V Port Profile Configuration Guide

Cisco Nexus 1000V Quality of Service Configuration Guide

Cisco Nexus 1000V REST API Plug-in Configuration Guide

Cisco Nexus 1000V Security Configuration Guide

Cisco Nexus 1000V System Management Configuration Guide

Cisco Nexus 1000V vCenter Plugin Configuration Guide

Cisco Nexus 1000V VXLAN Configuration Guide

Cisco Nexus 1000V VDP Configuration Guide

Cisco Nexus 1000V DFA Configuration Guide

Cisco Nexus 1000V vCenter Plugin Configuration Guide

Programming Guide

Cisco Nexus 1000V XML API User Guide

Reference Guides

Cisco Nexus 1000V Command Reference

Cisco Nexus 1000V Resource Availability Reference

Troubleshooting, Password Recovery, System Messages Guides

Cisco Nexus 1000V Troubleshooting Guide

Cisco Nexus 1000V Password Recovery Guide

Cisco NX-OS System Messages Reference

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/ps13095/tsd_products_support_series_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, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation* at: http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html.

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation, as an RSS feed and deliver content directly to your desktop using a reader application. The RSS feeds are a free service.

Overview

This chapter introduces the basic concepts, methodology, and general troubleshooting guidelines for problems that might occur when configuring and using the Cisco Nexus 1000V.

This chapter includes the following sections:

Troubleshooting Process

To troubleshoot your network, follow these steps:

Step 1 Gather information that defines the specific symptoms.

Step 2 Identify all potential problems that could be causing the symptoms.

Step 3 Systematically eliminate each potential problem (from most likely to least likely) until the symptoms disappear.

Best Practices

We recommend that you do the following to ensure the proper operation of your networks:

- Maintain a consistent Cisco Nexus 1000V release across all network devices.
- Refer to the release notes for your Cisco Nexus 1000V release for the latest features, limitations, and caveats.
- Enable system message logging. See the "Overview of Symptoms" section on page 1-4.

- Verify and troubleshoot any new configuration changes after implementing the change.

Troubleshooting Basics

This section introduces questions to ask when troubleshooting a problem with the Cisco Nexus 1000V or connected devices. Use the answers to these questions to identify the scope of the problem and to plan a course of action.

This section includes the following topics:

- Troubleshooting Guidelines, page 1-2
- Gathering Information, page 1-2
- Verifying Ports, page 1-3
- Verifying Layer 2 Connectivity, page 1-3
- Verifying Layer 3 Connectivity, page 1-3

Troubleshooting Guidelines

By answering the questions in the following subsections, you can determine the paths that you need to follow and the components that you should investigate further.

Answer the following questions to determine the status of your installation:

- Is this a newly installed system or an existing installation? (It could be a new host, switch, or VLAN).
- Has the host ever been able to see the network?
- Are you trying to solve an existing application problem (too slow, high latency, excessively long response time) or did the problem show up recently?
- What changed in the configuration or in the overall infrastructure immediately before the applications started to have problems?

To discover a network problem, follow these teps:

Step 1 Gather information on problems in your system. See the "Gathering Information" section on page 1-2.

Step 2 Verify the Layer 2 connectivity. See the "Verifying Layer 2 Connectivity" section on page 1-3.

Step 3 Verify the configuration for your end devices (storage subsystems and servers).

Step 4 Verify end-to-end connectivity. See the "Verifying Layer 3 Connectivity" section on page 1-3.

Gathering Information

This section highlights the tools that are commonly used to troubleshoot problems within your network. These tools are a subset of what you might use to troubleshoot your specific problem.

Each chapter in this guide includes additional tools and commands that are specific to the symptoms and possible problems covered in that chapter.

You should also have an accurate topology of your network to help isolate problem areas.

Use the following commands and examine the outputs:

- **show module**
- **show version**
- **show running-config**
- **show logging log**
- **show interfaces brief**
- **show vlan**
- **show accounting log**
- **show tech support svs**

Note To use commands with the **internal** keyword, you must log in with the network-admin role.

Verifying Ports

Answer the following questions to verify ports:

- Are you using the correct media copper or optical fiber type.
- Is the media broken or damaged?
- Are you checking a virtual Ethernet port? If so, use the **show interface brief** command. The status should be up.
- Are you checking a physical Ethernet port? If so, you need to check it by looking at the server or by looking at an upstream switch.
- Check if the network adapters of the Virtual Supervisor Module (VSM) virtual machine (VM) are assigned the right port groups and if all of them are connected from vSphere Client.

Verifying Layer 2 Connectivity

Answer the following questions to verify Layer 2 connectivity:

- Are the necessary interfaces in the same VLANs?
- Are all ports in a port channel configured the same for speed, duplex, and trunk mode?

Use the **show vlan brief** command. The status should be up.

Use the **show port-profile** command to check a port profile configuration.

Use the **show interface-brief** command to check the status of a virtual Ethernet port or a physical Ethernet port.

Verifying Layer 3 Connectivity

Answer the following questions to verify Layer 3 connectivity:

- Have you configured a gateway of last resort?

- Are any IP access lists, filters, or route maps blocking route updates?

Use the **ping** or **trace** commands to verify connectivity. See the following for more information:

- Ping, page 2-1
- Traceroute, page 2-2

Overview of Symptoms

The symptom-based troubleshooting approach provides multiple ways to diagnose and resolve problems. By using multiple entry points with links to solutions, this guide best serves users who may have identical problems that are perceived by different indicators. Search this guide in PDF form, use the index, or rely on the symptoms and diagnostics listed in each chapter as entry points to access necessary information in an efficient manner.

Using a given a set of observable symptoms on a network, it is important to be able to diagnose and correct software configuration issues and inoperable hardware components so that the problems are resolved with minimal disruption to the network. Those problems and corrective actions include the following:

- Identify key Cisco Nexus 1000V troubleshooting tools.
- Obtain and analyze protocol traces using SPAN or Ethanalyzer on the CLI.
- Identify or rule out physical port issues.
- Identify or rule out switch module issues.
- Diagnose and correct Layer 2 issues.
- Diagnose and correct Layer 3 issues.
- Obtain core dumps and other diagnostic data for use by the Technical Assistance Center (TAC).
- Recover from switch upgrade failures.

System Messages

The system software sends the syslog (system) messages to the console (and, optionally, to a logging server on another system) during operation. Not all messages indicate a problem with your system. Some messages are purely informational, while others might help diagnose problems with links, internal hardware, or the system software.

This section contains the following topics:

- System Message Text, page 1-4
- syslog Server Implementation, page 1-5

System Message Text

Message-text is a text string that describes the condition. This portion of the message might contain detailed information about the event, including terminal port numbers, network addresses, or addresses that correspond to locations in the system memory address space. Because the information in these variable fields changes from message to message, it is represented here by short strings enclosed in square brackets. A decimal number, for example, is represented as [dec].

```
2009 Apr 29 12:35:51 switch %KERN-1-SYSTEM_MSG: stun_set_domain_id : Setting domain ID
(1024) - kernel
```

Use this string to find the matching system message in the *Cisco NX-OS System Messages Reference System Messages Reference.*

Each system message is followed by an explanation and recommended action. The action may be as simple as “No action required.” It may involve a fix or a recommendation to contact technical support as shown in the following example:

Error Message `2009 Apr 29 14:57:23 switch %MODULE-5-MOD_OK: Module 3 is online (serial: )`

Explanation VEM module inserted successfully on slot 3.

Recommended Action None. This is an information message. Use the **show module** command to verify the module in slot 3.

syslog Server Implementation

The syslog facility allows the Cisco Nexus 1000V to send a copy of the message log to a host for more permanent storage. This feature can be useful if the logs need to be examined over a long period of time or when the Cisco Nexus 1000V is not accessible.

This example demonstrates how to configure a Cisco Nexus 1000V to use the syslog facility on a Solaris platform. Although a Solaris host is being used, the syslog configuration on all UNIX and Linux systems is very similar.

Syslog uses the concept of a facility to determine how it should be handled on the syslog server (the Solaris system in this example), and the message severity. Therefore, different message severities can be handled differently by the syslog server. They could be logged to different files or emailed to a particular user. Specifying a severity determines that all messages of that level and greater severity (lower number) will be acted upon.

Note The Cisco Nexus 1000V messages should be logged to a different file from the standard syslog file so that they cannot be confused with other non-Cisco syslog messages. The logfile should not be located on the / file system, to prevent log messages from filling up the / file system.
Syslog Client: switch1
Syslog Server: 172.22.36.211 (Solaris)
Syslog facility: local1
Syslog severity: notifications (level 5, the default)
File to log Cisco Nexus 1000V messages to: /var/adm/nxos_logs

To configure a syslog server, follow these steps:

Step 1 Configure the Cisco Nexus 1000V.

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch (config)# logging server 192.0.2.1 6 facility local1
```

Display the configuration.

```
switch# show logging server
```

```
Logging server: enabled
{192.0.2.1}
     server severity: notifications
     server facility: local1
```

Step 2 Configure the syslog server.

a. Modify /etc/syslog.conf to handle local1 messages. For Solaris, t at least one tab needs to be between the facility.severity and the action (/var/adm/nxos_logs).

```
#Below is for the NX-OS logging
local1.notice /var/adm/nxos_logs
```

b. Create the log file.

```
#touch /var/adm/nxos_logs
```

c. Restart the syslog.

```
# /etc/init.d/syslog stop
# /etc/init.d/syslog start
syslog service starting.
```

d. Verify that the syslog has started.

```
# ps -ef |grep syslogd
     root 23508 1 0 11:01:41 ? 0:00 /usr/sbin/syslogd
```

Step 3 Test the syslog server by creating an event in the Cisco Nexus 1000V. In this case, port e1/2 was bounced and the following was listed on the syslog server. Notice that the IP address of the switch is listed in brackets.

```
# tail -f /var/adm/nxos_logs
Sep 17 11:07:41 [172.22.36.142.2.2] : 2004 Sep 17 11:17:29 pacific:
%PORT-5-IF_DOWN_INITIALIZING: %$VLAN 1%$ Interface e 1/2 is down (Initializing)
Sep 17 11:07:49 [172.22.36.142.2.2] : 2004 Sep 17 11:17:36 pacific: %PORT-5-IF_UP:
%$VLAN 1%$ Interface e 1/2 is up in mode access
Sep 17 11:07:51 [172.22.36.142.2.2] : 2004 Sep 17 11:17:39 pacific:
%VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from pts/0
(dhcp-171-71-49-125.cisco.com)
```

Troubleshooting with Logs

The Cisco Nexus 1000V generates many types of system messages on the switch and sends them to a syslog server. These messages can be viewed to determine what events might have led up to the current problem condition that you are facing.

Viewing Logs

Use the following commands to access and view logs in the Cisco Nexus 1000V.

```
switch# show logging ?

console       Show console logging configuration
info          Show logging configuration
internal      syslog syslog internal information
last          Show last few lines of logfile
level         Show facility logging configuration
logfile       Show contents of logfile
```

```
loopback      Show logging loopback configuration
module        Show module logging configuration
monitor       Show monitor logging configuration
nvram         Show NVRAM log
pending       server address pending configuration
pending-diff  server address pending configuration diff
server        Show server logging configuration
session       Show logging session status
status        Show logging status
timestamp     Show logging timestamp configuration
|             Pipe command output to filter
```

Example 1-1 shows an example of the **show logging** command output.

Example 1-1 show logging Command

```
switch# show logging server
Logging server: enabled
{192.0.1.1}
server severity: critical
server facility: user
```

Cisco Support Communities

For additional information, visit one of the following support communities:

- Cisco Support Community for Server Networking
- Cisco Communities: Nexus 1000V

Contacting Cisco or VMware Customer Support

If you are unable to solve a problem after using the troubleshooting suggestions in this guide, contact a customer service representative for assistance and further instructions. Before you call, have the following information ready to help your service provider assist you as quickly as possible:

- Version of the Cisco Nexus 1000V software that you are running
- Version of the VMware ESX and vCenter Server software that you are running
- Contact phone number.
- Brief description of the problem
- Brief explanation of the steps that you have already taken to isolate and resolve the problem

If you purchased the Cisco Nexus 1000V and support contract from Cisco, contact Cisco for Cisco Nexus 1000V support. Cisco provides Layer 1, Layer 2, and Layer 3 support.

If you purchased the Cisco Nexus 1000V and an SNS through VMware, you should call VMware for Cisco Nexus 1000V support. VMware provides Layer 1 and Layer 2 support. Cisco provides Layer 3 support.

After you have collected this information, see the "Obtaining Documentation and Submitting a Service Request" section on page -xx.

For more information on the steps to take before calling Technical Support, see the "Gathering Information" section on page 1-2.

Troubleshooting Tools

This chapter describes the troubleshooting tools available for the Cisco Nexus 1000V and includes the following topics:

Commands

You use the command line interface (CLI) from a local console or remotely using a Telnet or Secure Shell SSH session. The CLI provides a command structure similar to Cisco NX-OS software, with context-sensitive help, **show** commands, multi-user support, and role-based access control.

Each feature has **show** commands that provide information about the feature configuration, status, and performance. Additionally, you can use the following commands for more information:

- **show system—**Provides information on system-level components, including cores, errors, and exceptions. Use the **show system error-id** command to find details on error codes:

```
switch# copy running-config startup-config
[########################################] 100%
2008 Jan 16 09:59:29 zoom %$ VDC-1 %$ %BOOTVAR-2-AUTOCOPY_FAILED: Autocopy of file
/bootflash/n1000-s1-dk9.4.0.0.837.bin.S8 to standby failed, error=0x401e0008

switch# show system error-id 0x401e0008
Error Facility: sysmgr
Error Description: request was aborted, standby disk may be full
```

Ping

The ping utility generates a series of *echo* packets to a destination across a TCP/IP internetwork. When the echo packets arrive at the destination, they are rerouted and sent back to the source. Using ping, you can verify connectivity and latency to a particular destination across an IP-routed network.

The ping utility allows you to ping a port or end device. By specifying the IPv4 address, you can send a series of frames to a target destination. Once these frames reach the target, they are looped back to the source and a time stamp is taken. Ping helps you to verify the connectivity and latency to destination.

Traceroute

Use traceroute to do the following:

- Trace the route followed by data traffic.
- Compute interswitch (hop-to-hop) latency.

Traceroute identifies the path taken on a hop-by-hop basis and includes a time stamp at each hop in both directions. You can use traceroute to test the connectivity of ports along the path between the generating switch and the switch closest to the destination.

Use the **traceroute** CLI command to access this feature.

If the destination cannot be reached, the path discovery starts, which traces the path up to the point of failure.

Monitoring Processes and CPUs

There CLI enables you to for monitor switch processes. CPU status, and utilization.

This section contains the following topics:

- Identifying the Running Processes and their States, page 2-2
- Displaying CPU Utilization, page 2-3
- Displaying CPU and Memory Information, page 2-4

Identifying the Running Processes and their States

Use the **show processes command** to identify the processes that are running and the status of each process. (See Example 2-1.) The command output includes the following:

- PID—Process ID.
- State —Process state.
- PC—Current program counter in hex format.
- Start_cnt—How many times a process has been started (or restarted).
- TTY—Terminal that controls the process. A “-” usually means a daemon is not running on any particular TTY.
- Process—Name of the process.

Process states are as follows:

- D—Uninterruptible sleep (usually I/O).
- R—Runnable (on run queue).
- S—Sleeping.
- T—Traced or stopped.

- Z—Defunct (“zombie”) process.
- NR—Not running.
- ER—Should be running but is currently not running.

Note The ER state typically designates a process that has been restarted too many times, causing the system to classify it as faulty and disable it.

Example 2-1 show processes Command

```
switch# show processes ?
  cpu     Show processes CPU Info
  log     Show information about process logs
  memory  Show processes Memory Info

switch# show processes

PID    State  PC        Start_cnt    TTY   Process
-----  -----  --------  -----------  ----  -------------
    1      S  b7f9e468            1     -  init
    2      S         0            1     -  migration/0
    3      S         0            1     -  ksoftirqd/0
    4      S         0            1     -  desched/0
    5      S         0            1     -  migration/1
    6      S         0            1     -  ksoftirqd/1
    7      S         0            1     -  desched/1
    8      S         0            1     -  events/0
    9      S         0            1     -  events/1
   10      S         0            1     -  khelper
   15      S         0            1     -  kthread
   24      S         0            1     -  kacpid
  101      S         0            1     -  kblockd/0
  102      S         0            1     -  kblockd/1
  115      S         0            1     -  khubd
  191      S         0            1     -  pdflush
  192      S         0            1     -  pdflushn
...
```

Displaying CPU Utilization

Use the **show processes cpu** command to display CPU utilization. See Example 2-2. The command output includes the following:

- Runtime(ms)—CPU time the process has used, expressed in milliseconds.
- Invoked—Number of times the process has been invoked.
- uSecs—Microseconds of CPU time in average for each process invocation.
- 1Sec—CPU utilization in percentage for the last one second.

Example 2-2 show processes cpu Command

```
switch# show processes cpu

PID    Runtime(ms)  Invoked   uSecs  1Sec   Process
-----  -----------  --------  -----  -----  -----------
```

```
    1          922  4294967295      0      0  init
    2          580    377810      1      0  migration/0
    3          889   3156260      0      0  ksoftirqd/0
    4         1648    532020      3      0  desched/0
    5          400    150060      2      0  migration/1
    6         1929   2882820      0      0  ksoftirqd/1
    7         1269    183010      6      0  desched/1
    8         2520  47589180      0      0  events/0
    9         1730   2874470      0      0  events/1
   10           64    158960      0      0  khelper
   15            0    106970      0      0  kthread
   24            0     12870      0      0  kacpid
  101           62   3737520      0      0  kblockd/0
  102           82   3806840      0      0  kblockd/1
  115            0     67290      0      0  khubd
  191            0      5810      0      0  pdflush
  192          983   4141020      0      0  pdflush
  194            0      5700      0      0  aio/0
  193            0      8890      0      0  kswapd0
  195            0      5750      0      0  aio/1
...
```

Displaying CPU and Memory Information

Use the **show system resources** command to display system-related CPU and memory statistics. See Example 2-3. The output includes the following:

- Load average is defined as the number of running processes. The average reflects the system load over the past 1, 5, and 15 minutes.
- Processes is the number of processes in the system, and how many are actually running when the command is issued.
- CPU states is the CPU usage percentage in user mode, kernel mode, and idle time in the last one second.
- Memory usage provides the total memory, used memory, free memory, memory used for buffers, and memory used for the cache in KB. Buffers and cache are also included in the used memory statistics.

Example 2-3 show system resources Command

```
switch# show system resources
Load average:   1 minute: 0.30   5 minutes: 0.34   15 minutes: 0.28
Processes   :   606 total, 2 running
CPU states  :   0.0% user,   0.0% kernel,   100.0% idle
Memory usage:   2063268K total,   1725944K used,    337324K free
                   2420K buffers,  857644K cache
```

RADIUS

RADIUS is a protocol used for the exchange of attributes or credentials between a head-end RADIUS server and a client device. These attributes relate to three classes of services:

- Authentication
- Authorization

- Accounting

Authentication refers to the authentication of users for access to a specific device. You can use RADIUS to manage user accounts for access to a Cisco Nexus 1000V. When you try to log into a device, the Cisco Nexus 1000V validates you with information from a central RADIUS server.

Authorization refers to the scope of access that you have once you have been authenticated. Assigned roles for users can be stored in a RADIUS server with a list of actual devices that the user should have access to. Once the user has been authenticated, the switch can then refer to the RADIUS server to determine the extent of access the user will have within the switch network.

Accounting refers to the log information that is kept for each management session in a switch. This information can be used to generate reports for troubleshooting purposes and user accountability. Accounting can be implemented locally or remotely (using RADIUS).

The following is an example of an accounting log entries:

```
switch# show accounting log
Sun Dec 15 04:02:27 2002:start:/dev/pts/0_1039924947:admin
Sun Dec 15 04:02:28 2002:stop:/dev/pts/0_1039924947:admin:vsh exited normally
Sun Dec 15 04:02:33 2002:start:/dev/pts/0_1039924953:admin
Sun Dec 15 04:02:34 2002:stop:/dev/pts/0_1039924953:admin:vsh exited normally
Sun Dec 15 05:02:08 2002:start:snmp_1039928528_172.22.95.167:public
Sun Dec 15 05:02:08 2002:update:snmp_1039928528_172.22.95.167:public:Switchname
```

Note The accounting log shows only the beginning and ending (start and stop) for each session.

Syslog

The system message logging software saves messages in a log file or directs the messages to other devices. This feature provides the following capabilities:

- Logging information for monitoring and troubleshooting.
- Selection of the types of logging information to be captured.
- Selection of the destination of the captured logging information.

Syslog allows you to store a chronological log of system messages locally or sent to a central syslog server. Syslog messages can also be sent to the console for immediate use. These messages can vary in detail depending on the configuration that you choose.

Syslog messages are categorized into seven severity levels from *debug to critical* events. You can limit the severity levels that are reported for specific services within the switch.

Log messages are not saved across system reboots. However, a maximum of 100 log messages with a severity level of critical and below (levels 0, 1, and 2) can logged to a local file or server.

Logging Levels

The Cisco Nexus 1000V supports the following logging levels:

- 0—emergency
- 1—alert
- 2—critical
- 3—error

- 4—warning
- 5—notification
- 6—informational
- 7—debugging

By default, the switch logs normal but significant system messages to a log file and sends these messages to the system console. Users can specify which system messages should be saved based on the type of facility and the severity level. Messages are time-stamped to enhance real-time debugging and management.

Enabling Logging for Telnet or SSH

System logging messages are sent to the console based on the default or configured logging facility and severity values.

Users can disable logging to the console or enable logging to a given Telnet or SSH session.

- To disable console logging, use the **no logging console** command in global CONFIGURATION mode.
- To enable logging for Telnet or SSH, use the **terminal monitor** command in EXEC mode.

Note Note: When logging to a console session is disabled or enabled, that state is applied to all future console sessions. If a user exits and logs in again to a new session, the state is preserved. However, when logging to a Telnet or SSH session is enabled or disabled, that state is applied only to that session. The state is not preserved after the user exits the session.

The **no logging console** command shown in Example 2-4 does the following:

- Disables console logging
- Enabled by default

Example 2-4 no logging console Command

```
switch(config)# no logging console
```

The **terminal monitor** command shown in Example 2-5 does the following:

- Enables logging for Telnet or SSH
- Disabled by default

Example 2-5 terminal monitor Command

```
switch# terminal monitor
```

For more information about configuring syslog, see the *Cisco Nexus 1000V System Management Configuration Guide*.

CHAPTER 3

Installation

This chapter describes how to identify and resolve installation problems and includes the following topics:

Isolating Installation Problems

This section explains how to isolate possible installation problems.

Verifying Your VMware License Version

Before you begin to troubleshoot any installation issues, you should verify that your ESX server has the VMware Enterprise Plus license that includes the Distributed Virtual Switch feature.

BEFORE YOU BEGIN

Before you begin, you must know or do the following:

- You are logged in to the vSphere client on the ESX server.
- You are logged in to the Cisco Nexus 1000V CLI in EXEC mode.
- This procedure verifies that your vSphere ESX server uses the VMware Enterprise Plus license. This license includes the Distributed Virtual Switch feature, which allows visibility to the Cisco Nexus 1000V.

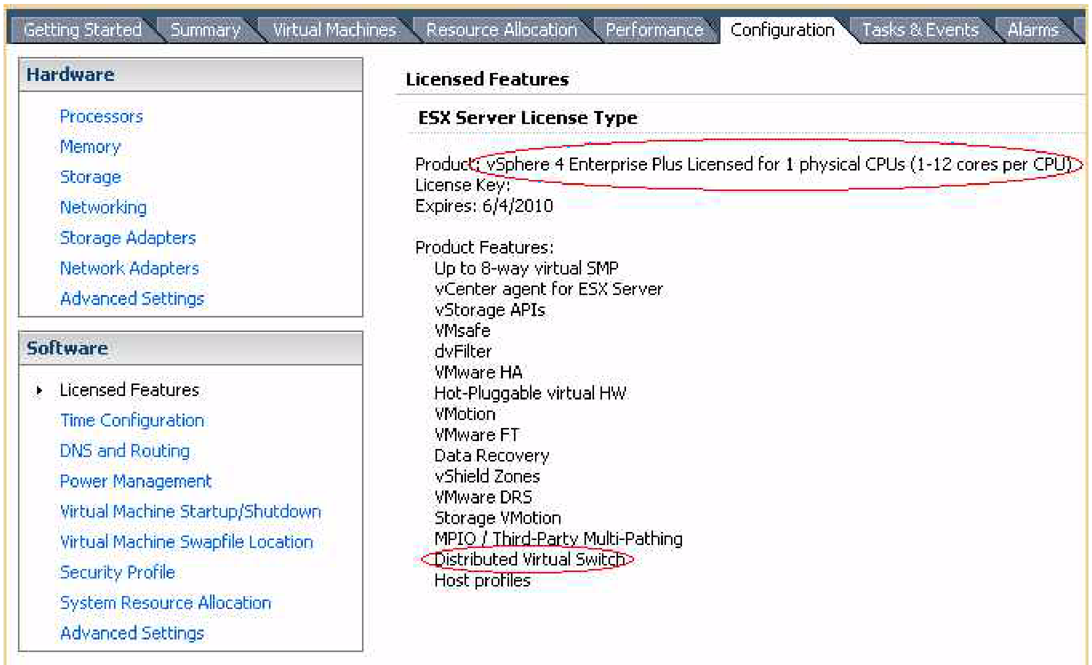
- If your vSphere ESX server does not have the Enterprise Plus license, then you must upgrade your license.

DETAILED STEPS

Step 1 From the vSphere Client, choose the host whose Enterprise Plus license you want to check.

Step 2 Click the **Configuration** tab and choose **Licensed Features**.

The Enterprise Plus licensed features are displayed.



Licensed Features

ESX Server License Type

Product: vSphere 4 Enterprise Plus Licensed for 1 physical CPUs (1-12 cores per CPU)
License Key:
Expires: 6/4/2010

Product Features:
Up to 8-way virtual SMP
vCenter agent for ESX Server
vStorage APIs
VMsafe
dvFilter
VMware HA
Hot-Pluggable virtual HW
VMotion
VMware FT
Data Recovery
vShield Zones
VMware DRS
Storage VMotion
MPIO / Third-Party Multi-Pathing
Distributed Virtual Switch
Host profiles

Step 3 Verify that the following are included in the Licensed Features:

- Enterprise Plus license
- Distributed Virtual Switch feature

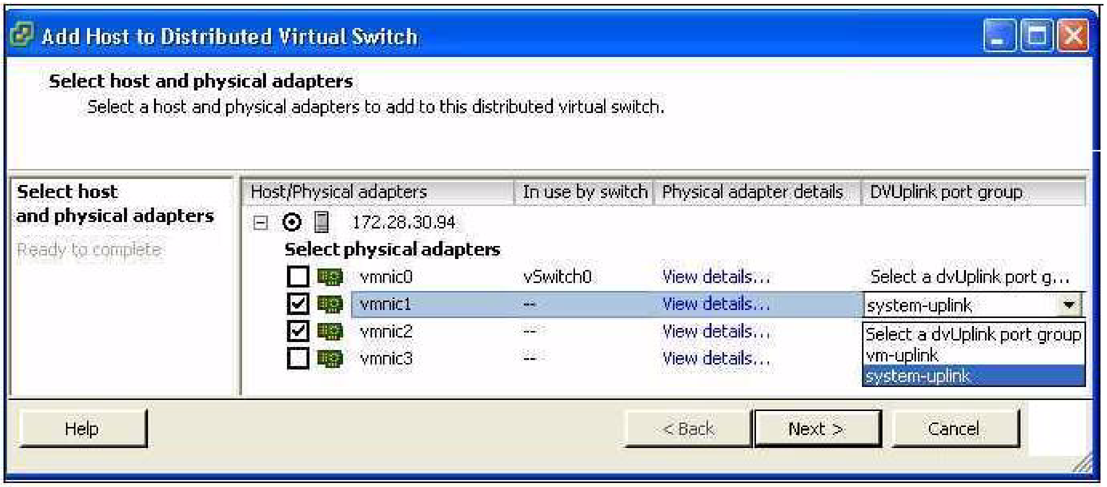
Step 4 Do one of the following:

- If your vSphere ESX server has an Enterprise Plus license, you have the correct license and visibility to the Cisco Nexus 1000V.
- If your vSphere ESX server does not have an Enterprise Plus license, you must upgrade your VMware License to an Enterprise Plus license to have visibility to the Cisco Nexus 1000V.

Host is Not Visible from the Distributed Virtual Switch

If you have added hosts and adapters with your VSM, you must also add them in the vCenter Client Add Host to Distributed Virtual Switch dialog box shown in Figure 3-1.

Figure 3-1 ***Host is Visible from the Distributed Virtual Switch***



Add Host to Distributed Virtual Switch

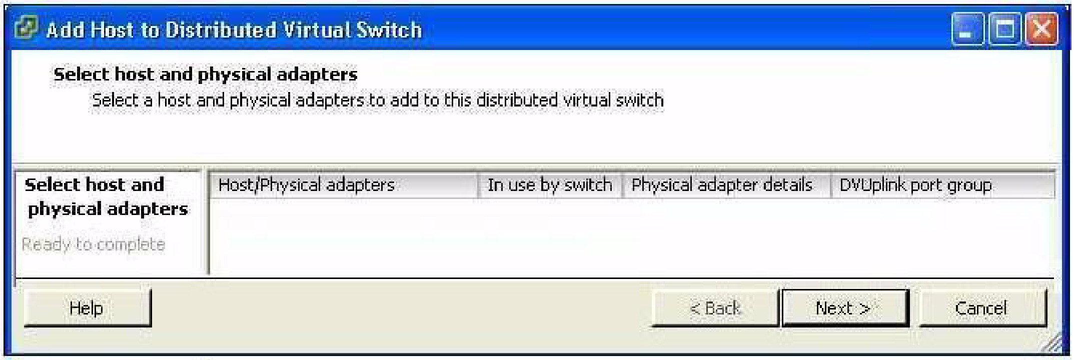
Select host and physical adapters
Select a host and physical adapters to add to this distributed virtual switch.

Host/Physical adapters	In use by switch	Physical adapter details	DVUplink port group
⊟ ⊙ 172.28.30.94			
Select physical adapters			
□ vmnic0	vSwitch0	View details...	Select a dvUplink port g...
☑ vmnic1	--	View details...	system-uplink
☑ vmnic2	--	View details...	Select a dvUplink port group vm-uplink system-uplink
□ vmnic3	--	View details...	

If the hosts and adapters do not appear in this dialog box, you might have the incorrect VMware license installed on your ESX server.

Use the "Verifying Your VMware License Version" procedure on page 3-1 to confirm.

Figure 3-2 ***Host is Not Visible from the Distributed Virtual Switch***



Add Host to Distributed Virtual Switch

Select host and physical adapters
Select a host and physical adapters to add to this distributed virtual switch

Host/Physical adapters	In use by switch	Physical adapter details	DVUplink port group

Refreshing the vCenter Server Connection

You can refresh the connection between the Cisco Nexus 1000V and vCenter Server.

Step 1 From the Cisco Nexus 1000V Connection Configuration mode on the Virtual Supervisor Module (VSM), enter the following command sequence:

```
Example:
switch# config t
switch(config)# svs connection s1
switch(config-svs-conn)# no connect
switch(config-svs-conn)# connect
```

Step 2 You have completed this procedure.

Improving Performance on the ESX and VM

Use the following pointers to improve performance on the ESX host and the VMs.

- Install VMware Tools on the vCenter Server VM, with Hardware Acceleration enabled.
- Use the command line interface in the VMs instead of the graphical interface where possible.

Verifying the Domain Configuration

The Virtual Supervisor Module (VSM) and Virtual Ethernet Module (VEM) are separated within a Layer 2 domain. To allow VSM-VEM pairs to communicate within the same Layer 2 domain, each pair must have a unique identifier. The domain ID serves as the unique identifier that allows multiple VSM-VEM pairs to communicate inside the same Layer 2 domain.

Following the installation of the Cisco Nexus 1000V, make certain that you configure a domain ID. Without a domain ID, the VSM cannot connect to the vCenter Server. Follow these guidelines:

- The domain ID should be a value within the range of 1 to 4095.
- All the control traffic between the VSM and the VEM is carried over the configured control VLAN.
- All the data traffic between the VSM and the VEM is carried over the configured packet VLAN.
- Make sure that the control VLAN and the packet VLAN are allowed on the port in the upstream switch to which the physical NIC of the host hosting the VSM and VEM VM are connected.

Verifying the Port Group Assignments for a VSM VM Virtual Interface

You can verify that two port groups are created on the ESX hosting the VSM VM through the vCenter Server. The following port groups (PG) should be created:

- Control PG (Vlan = Control VLAN)
- Packet PG (Vlan = Packet VLAN)
- Management PG (Vlan = Management VLAN)

Make sure the port groups are assigned to the three virtual interfaces of the VSM VM in the following order:

Virtual Interface Number	Port Group
Network Adapter 1	Control PG
Network Adapter 2	MGMT PG
Network Adapter 3	Packet PG

To verify if the VSM VM network adapter 1, network adapter 2, and network adapter 3 are carrying the control VLAN, management VLAN, and the packet VLAN, follow these steps:

Step 1 Enter the **show mac address-table dynamic interface vlan** *control-vlan* command on the upstream switch.

Expected output: the network adapter1 MAC address of the VSM VM.

Step 2 Enter the **show mac address-table dynamic interface vlan** *mgmt-vlan* command on the upstream switch.

Expected output: the network adapter2 MAC address of the VSM VM.

Step 3 Enter the **show mac address-table dynamic interface vlan** *packet-vlan* command on the upstream switch.

Expected output: the network adapter3 MAC address of the VSM VM.

Verifying VSM and vCenter Server Connectivity

When troubleshooting connectivity between the VSM and vCenter Server, follow these guidelines:

- Make sure that domain parameters are configured correctly.
- Make sure the Windows VM hosting the vCenter Server has the following ports open.
  - Port 80
  - Port 443
- Try reloading the VSM if after verifying the preceding steps, the connect still fails.
- Check if the VSM extension is created by the vCenter Server by pointing your web browser to https://your-virtual-center/mob/, and choosing **Content > Extension Manager**.

Step 1 Ensure that the Nexus N1000V VSM VM network adapters are configured properly.

Step 2 Make sure that the Windows VM machine hosting the vCenter Server has the following ports open:

- Port 80
- Port 443

Step 3 Ping the vCenter Server from the Cisco Nexus 1000V VSM.

Step 4 Ensure that the VMware VirtualCenter Server service is running.

Troubleshooting Connections to vCenter Server

You can troubleshoot connections between a Cisco Nexus 1000V VSM and a vCenter Server.

Step 1 In a web browser, enter the path: http://<VSM-IP>

Step 2 Download the cisco_nexus_1000v_extension.xml file to your desktop.

Step 3 From the vCenter Server menu, choose **Plugins > Manage Plugins.** Right click an empty area and select the plugin in Step2 as the New Extension.

If these steps fail, you might be using an out-of-date.xml file.

Confirm that the extension is available:

Step 1 In a web browser, enter the path: http://<vCenter-Server-IP>/mob.

Step 2 Click **Content**.

Step 3 Click **extensionManager**.

Step 4 If extensionList[Cisco_Nexus_1000v_584325821] is displayed in the value column, proceed to connect to the VSM.

Note The actual value of "Cisco_Nexus_1000V_584325821" will vary. It should match the extension key from the cisco_nexus_1000v_extension.xml file.

Recovering the Network Administrator Password

For information about recovering the network administrator password, see the *Cisco Nexus 1000V Password Recovery Guide*.

Managing Extension Keys

This section includes the following topics:

- Known Extension Problems and Resolutions, page 3-7
- Resolving a Plug-In Conflict, page 3-7
- Finding the Extension Key on the Cisco Nexus 1000V, page 3-7
- Finding the Extension Key Tied to a Specific DVS, page 3-8
- Verifying Extension Keys, page 3-8

Known Extension Problems and Resolutions

Use the following table to troubleshoot and resolve known problems with plug-ins and extensions.

Problem	Resolution
The extension does not show up immediately in the plugin.	Close the VI client and then open the VI client again.
You cannot delete the extension from the VI client.	If you delete the extension using Manager Object Browser (MOB), the VI client screen might not refresh and indicate that the extension was deleted. In this case, close the VI client and then open the VI client again.
If you click the **download and install** link for the extension. you see an invalid URI.	None.You do not need to click **download and install**. If you do, it has no effect on the installation or connectivity. The plug-in only needs to be registered with vCenter.

Resolving a Plug-In Conflict

If you see “The specified parameter was not correct,” when Creating a Nexus 1000V plug-in on vCenter Server, you have tried to register a plug-in that is already registered.

Use the following procedure to resolve this problem.

Step 1 Make sure that you are using the correct cisco_nexus1000v_extension.xml file.

Step 2 Make sure that you have refreshed your browser because it caches this file and unless refreshed it might cache obsolete content with the same filename.

Step 3 Follow the steps described in the “Verifying Extension Keys” section on page 3-8 to compare the extension key installed on the VSM with the plug-in installed on the vCenter Server.

Finding the Extension Key on the Cisco Nexus 1000V

You can find the extension key on the Cisco Nexus 1000V.

BEFORE YOU BEGIN

- Log in to the Cisco Nexus 1000V VSM CLI in EXEC mode.
- Know that you can use the extension key in the “Unregistering the Extension Key in the vCenter Server” section on page 3-11.

DETAILED STEPS

Step 1 From the Cisco Nexus 1000V for the VSM whose extension key you want to view, enter the following command:

show vmware vc extension-key

```
Example:
switch# show vmware vc extension-key
```

```
Extension ID: Cisco_Nexus_1000V_1935882621
switch#
```

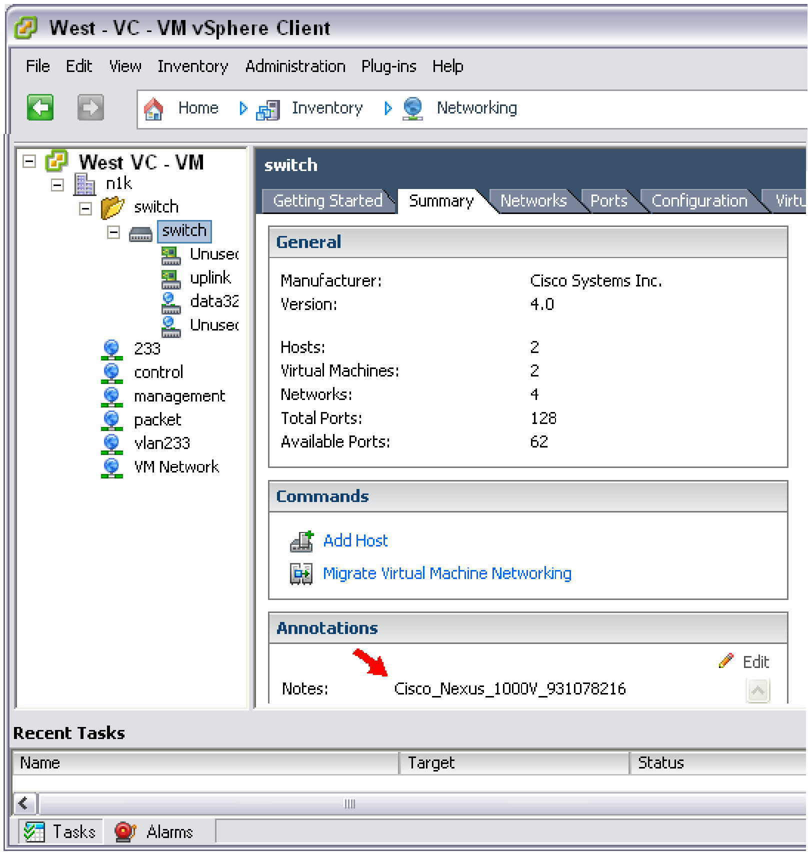
Finding the Extension Key Tied to a Specific DVS

You can find the extension key tied to a specific DVS.

Step 1 From the vSphere Client, choose the DVS whose extension key you want to find.

Step 2 Click the **Summary** tab.

The Summary tab opens with the extension key displayed in the Notes section of the Annotations block.



General

Manufacturer: Cisco Systems Inc.
Version: 4.0

Hosts: 2
Virtual Machines: 2
Networks: 4
Total Ports: 128
Available Ports: 62

Annotations

Notes: Cisco_Nexus_1000V_931078216

Verifying Extension Keys

You can verify that the Cisco Nexus 1000V and vCenter Server are using the same extension key.

DETAILED STEPS

Step 1 Find the extension key used on the Cisco Nexus 1000V using the "Finding the Extension Key on the Cisco Nexus 1000V" section on page 3-7.

Step 2 Find the extension key used on the vCenter Server using the "Finding the Extension Key Tied to a Specific DVS" section on page 3-8.

Step 3 Verify that the two extension keys (the one found in Step 1 with that in Step 2) are the same.

Re-registering a new Cisco Nexus 1000V VSM with an old DVS Instance

You can create the complete Cisco Nexus 1000V configuration in case the existing VSM does not bootup to normal mode, or was deleted. In that case, you may take the backup of the old running configuration and restore by deploying a new VSM and attach it to the same DVS.

Steps to re-register a new VSM with an old DVS instance:

DETAILED STEPS

Step 1 Note down the N1000V Extension key from the DVS summary on the vCenter. Ex: Cisco_Nexus_1000V_17750897.

Step 2 Deploy a new VSM of the same version and with the same name as DVS.

```
Example:
switch(config)# switchname vsm_36
```

Step 3 Copy all the previous running configuration (except svs connection) to the new VSM.

Step 4 Unregister the old extension key from vCenter MOB. Ex: https://(*ip_address*)/mob. Go to **Content** > **Extension Manager** > **UnregisterExtension** and provide the old extension key and click **Invoke Method.** It must result in void.

Step 5 Change the extension key in the new VSM to the old value.

```
Example:
vsm_36(config)# vmware vc extension-key Cisco_Nexus_1000V_17750897
vsm_36(config)# end
vsm_36# show vmware vc extension-key
Extension ID: Cisco_Nexus_1000V_17750897
```

Step 6 Retrieve the old DVS uuid from vCenter MOB. Go to **rootFolder(group-d1)** > **childEntity** > **networkFolder** > **childEntity(old DVS obj)** > **childEntity(DVS obj)** > **uuid**. Note down the uuid string. Ex: 50 06 00 94 4f d0 e4 0d-4b 31 c6 4a 5a 40 70 9a

Step 7 Configure a new svs connection as shown in the example below.

```
Example:
vsm_36(config)# svs connection vc
vsm_36(config-svs-conn)# remote ip address ip_address
vsm_36(config-svs-conn)# protocol vmware-vim
vsm_36(config-svs-conn)# register-plugin remote username administrator@vsphere.local
password Secret@123
```

```
vsm_36(config-svs-conn)# vmware dvs uuid "50 06 00 94 4f d0 e4 0d-4b 31 c6 4a 5a 40 70 9a"
datacenter-name D
vsm_36(config-svs-conn)# connect
vsm_36(config-svs-conn)# end

Now, it should connect to vCenter with the old DVS.
```

Step 8 Check using the **show svs connection** command.

```
Example:
vsm_36# show svs connections
connection vc:
hostname: -
ip address: (as specified in step 4)
ipv6 address: -
remote port: 80
transport type: ipv4
protocol: vmware-vim https
certificate: default
datacenter name: D
admin:
max-ports: 12000
DVS uuid: 50 06 00 94 4f d0 e4 0d-4b 31 c6 4a 5a 40 70 9a
dvs version: 5.0.0
config status: Enabled
operational status: Connected
sync status: in progress
version: VMware vCenter Server 6.5.0 build-5973321
vc-uuid: 27230ef3-dfb0-4aa5-9af5-37ddef9418b4
ssl-cert: self-signed or not authenticated
```

Removing Hosts from the Cisco Nexus 1000V DVS

You can remove hosts from the Cisco Nexus 1000V DVS.

BEFORE YOU BEGIN

- Log in to vSphere Client.
- Know the name of the Cisco Nexus 1000V DVS to remove from vCenter Server.

DETAILED STEPS

Step 1 From vSphere Client, choose **Inventory > Networking**.

Step 2 Choose the DVS for the Cisco Nexus 1000V and click the **Hosts** tab.

The Host tab opens.

Step 3 Right-click each host, and choose **Remove from Distributed Virtual Switch**.

The hosts are now removed from the DVS.

Removing the Cisco Nexus 1000V from the vCenter Server

You can remove the Cisco Nexus 1000V DVS from vCenter Server.

BEFORE YOU BEGIN

- Log in to the VSM CLI in EXEC mode.

DETAILED STEPS

Step 1 From the Cisco Nexus 1000V VSM, use the following commands to remove the DVS from the vCenter Server.

a. **config t**

b. **svs connection vc**

c. **no vmware dvs**

```
Example:
switch# conf t
switch(config)# svs connection vc
switch(config-svs-conn)# no vmware dvs
switch(config-svs-conn)#
```

The DVS is removed from vCenter Server.

Step 2 You have completed this procedure.

Unregistering the Extension Key in the vCenter Server

You can unregister the Cisco Nexus 1000V extension key in vCenter Server.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

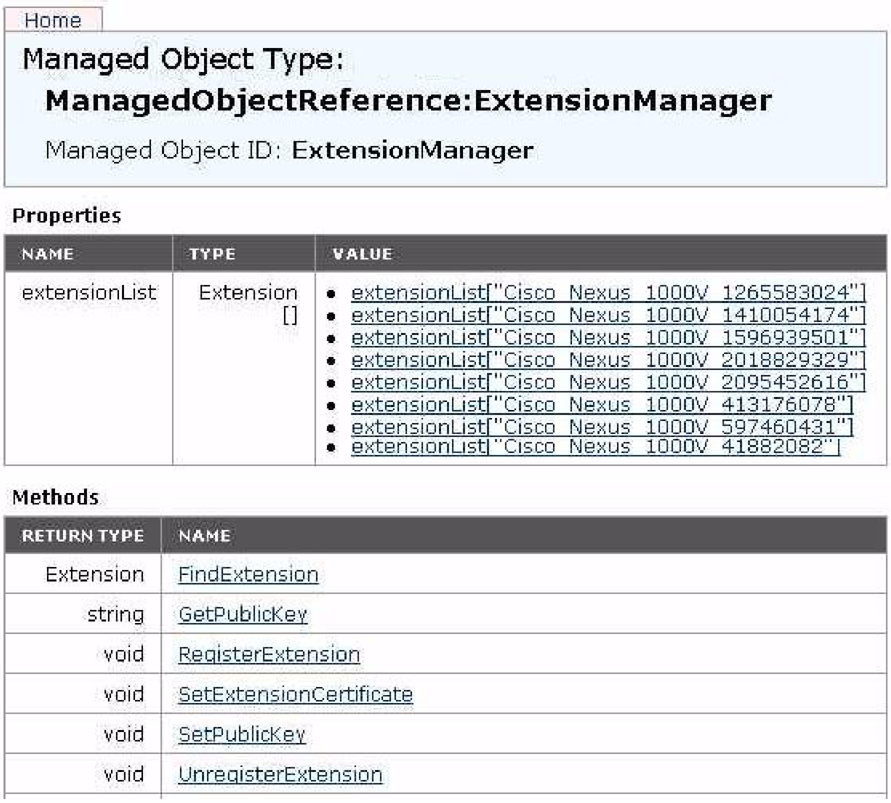
- Open a browser window.
- Paste the extension key name into the vCenter Server MOB. You should already have the extension key found in the "Finding the Extension Key on the Cisco Nexus 1000V" section on page 3-7.
- After unregistering the extension key in vCenter Server, you can start a new installation of the Cisco Nexus 1000V VSM software.

DETAILED STEPS

Step 1 Point your browser to the following URL:

https://<vc-ip>/mob/?moid=ExtensionManager

The Extension Manager opens in your Manager Object Browser (MOB).



Home

Managed Object Type: **ManagedObjectReference:ExtensionManager**

Managed Object ID: **ExtensionManager**

Properties

NAME	TYPE	VALUE
extensionList	Extension []	• extensionList["Cisco Nexus 1000V 1265583024"] • extensionList["Cisco Nexus 1000V 1410054174"] • extensionList["Cisco Nexus 1000V 1596939501"] • extensionList["Cisco Nexus 1000V 2018829329"] • extensionList["Cisco Nexus 1000V 2095452616"] • extensionList["Cisco Nexus 1000V 413176078"] • extensionList["Cisco Nexus 1000V 597460431"] • extensionList["Cisco Nexus 1000V 41882082"]

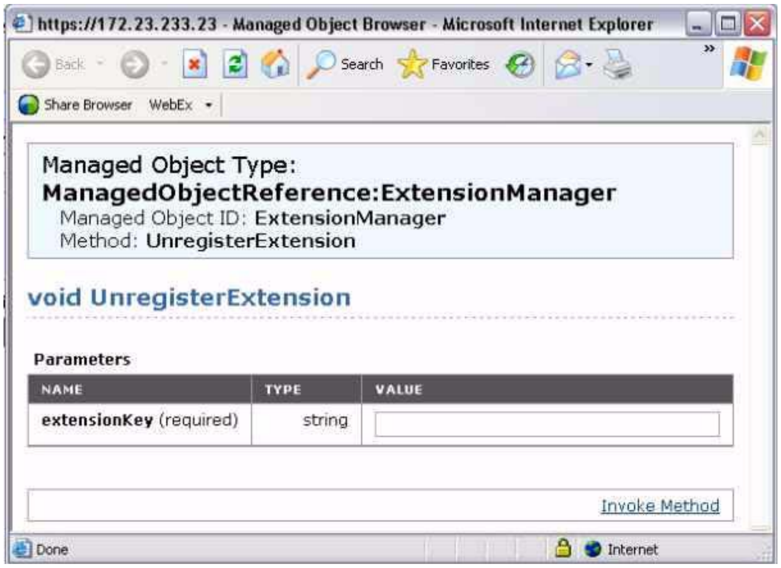
Methods

RETURN TYPE	NAME
Extension	FindExtension
string	GetPublicKey
void	RegisterExtension
void	SetExtensionCertificate
void	SetPublicKey
void	UnregisterExtension

Step 2 Click **Unregister Extension**.

https://<vc-ip>/mob/?moid=ExtensionManager&method=unregisterExtension

A dialog box opens to unregister the extension.



https://172.23.233.23 - Managed Object Browser - Microsoft Internet Explorer
Back
Search
Favorites
Share Browser
WebEx
Done
Internet

Managed Object Type:
ManagedObjectReference:ExtensionManager
Managed Object ID: **ExtensionManager**
Method: **UnregisterExtension**

void UnregisterExtension

Parameters

NAME	TYPE	VALUE
extensionKey (required)	string	

Invoke Method

Step 3 In the value field, paste the extension key that you found in the "Finding the Extension Key on the Cisco Nexus 1000V" section on page 3-7, and then click **Invoke Method**.

The extension key is unregistered in vCenter Server so that you can start a new installation of the Cisco Nexus 1000V VSM software.

Step 4 You have completed this procedure.

Problems with the Cisco Nexus 1000V Installation Management Center

The following are possible problems and their solutions.

.

Symptom	Problem	Recommended Action
Port migration fails.	The VSM to VEM migration fails in Layer 2 / Layer 3 mode installation.	• Check if there is any VM running on the vSwitch. You need to power off all VMs running on the vSwitch before migration. • Check if the vCenter is Virtual Update Manager (VUM) enabled. Before migration, the host is added to the DVS by using VUM. • Verify that the native VLAN in the upstream switch configuration is correct. • Ensure that the VUM repositories are up-to-date and accurate.
The VEM is missing on the VSM after the migration.	• The installer application finishes successfully with port migration in Layer 3 mode. • The VEM is added to the vCenter but does not display when the **show module** command is entered on the VSM.	• Verify that the Layer 3 control profile VLAN is configured as a system VLAN. • Verify that the uplink profile is allowing the Layer 3 control VTEP VLAN and that it is a system VLAN. • From the ESX host (VEM), enter a **vmkping** to the mgmt0/control0 IP address. It should be successful. If not, check the intermediate switches for proper routes between the subnets. • The VTEP should be pingable from the VSM. • Check the vCenter MOB for opaque data propagation.
Configuration file issue.	After loading the previously saved configuration file, the installation application does not complete.	• Check the configuration file for appropriate contents. **Note** You might need to change a few of the fields before reusing the previously saved files. • Check if a VM with the same name already exists in the DC. This can be identified by reviewing the Virtual Machine field in the configuration file.

Licenses

This chapter describes how to identify and resolve problems related to licenses and includes the following sections:

- Information About Licenses, page 4-1
- Prerequisites to License Troubleshooting, page 4-2
- Problems with Licenses, page 4-3
- License Troubleshooting Commands, page 4-4

Information About Licenses

The name for the Cisco Nexus 1000V license package is NEXUS1000V_LAN_SERVICES_PKG and the version is 3.0. By default, 1024 licenses are installed with the Virtual Supervisor Module (VSM). These default licenses are valid for 60 days. You can purchase permanent licenses that do not expire.

Licensing is based on the number of CPU sockets on the ESX servers attached as Virtual Ethernet Modules (VEM) to the VSM.

A module is either licensed or unlicensed:

- Licensed module—A VEM is licensed if it acquires licenses for all of its CPU sockets from the pool of available licenses installed on the VSM.
- Unlicensed module—A VEM is unlicensed if it does not acquire licenses for all of its CPU sockets from the pool of available licenses installed on the VSM.

  If a VEM is unlicensed, the virtual Ethernet ports correspond to the virtual machines (VMs) that are kept down and are shown as unlicensed.

Note The server administrator has no information about VEM licenses. The VEM licensed state must be communicated to server administrators so they are aware that vEthernet interfaces on unlicensed modules cannot pass traffic.

For additional information about licensing, including how to purchase, install, or remove an installed license, see the *Cisco Nexus 1000V License Configuration Guide*.

Contents of the License File

The contents of the Cisco Nexus 1000V license file indicates the number of licenses purchased and the host ID. To display the contents of a license file, use the **show license file** *license_name* command.

```
switch# show license file sample.lic
sample.lic:
SERVER this_host ANY
VENDOR cisco
INCREMENT NEXUS1000V_LAN_SERVICES_PKG cisco 1.0 permanent 16 \
        HOSTID=VDH=8449368321243879080 \
        NOTICE="<LicFileID>sample.lic</LicFileID><LicLineID>0</LicLineID> \
        <PAK>dummyPak</PAK>" SIGN=34FCB2B24AE8
```

The host ID that appears in the license file must match that shown on the VSM. To verify the match, use the **show license host-id** command. See Example 4-3 on page 4-6.

Caution Do not edit the contents of the license file. The license is invalidated if its contents are altered. If you have already done so, contact your Cisco Customer Support Account Team.

Prerequisites to License Troubleshooting

Before you begin troubleshooting licenses, verify the information in this checklist:

- Make sure that the name of the license file has fewer than 32 characters by using the **show license usage** command. See Example 4-1 on page 4-5.
- Make sure that no other license file with the same name is installed on the VSM by using the **show license usage** command. See Example 4-1 on page 4-5. If there is a license file with the same name, rename your new license file to something else.
- Do not edit the contents of the license file. If you have already done so, contact your Cisco Customer Support Account Team.
- Make sure that the host ID in the license file is the same as the host ID on the switch by using the **show license host-id** command and the **show license file** command. See Example 4-3 on page 4-6 and Example 4-4 on page 4-6.

Problems with Licenses

The following are symptoms, possible causes, and solutions for problems with licenses.

Symptom	Possible Causes	Solution
When you power on a virtual machine with ports on a Cisco Nexus 1000V port group, the interfaces do not come up, but display the following status: `VEM Unlicensed`	A license could not be obtained for the server (VEM) where the virtual machine resides.	1. Verify the license usage. **show license usage** *license_name* See Example 4-1 on page 4-5. 2. Determine the number of licenses required by viewing the sockets installed on the VEM. **show module vem license-info** See Example 4-8 on page 4-7. 3. Contact your Cisco Customer Support Account Team to acquire additional licenses.

You see the following system message:
`PLATFORM-2-PFM_LIC_WARN_EXP Syslog`
`2008 Dec 19 22:28:30 N1KV %PLATFORM-2-PFM_LIC_WARN_EXP: WARNING License for VEMs is about to expire in 1 days! The VEMs' VNICS will be brought down if license is allowed to expire. Please contact your Cisco account team or partner to purchase Licenses. To activate your purchased licenses, click on www.cisco.com/go/license.`
	The default or evaluation license in use is about to expire.
	Note Permanent licenses do not expire.
		1. Verify the license usage.
		show license usage *license_name*
		See Example 4-1 on page 4-5.
		2. Contact your Cisco Customer Support Account Team to acquire additional licenses.

Symptom	Possible Causes	Solution
You see the following system message: `%LICMGR-2-LOG_LIC_USAGE: Feature NEXUS1000V_LAN_SERVICES_PKG is using 17 licenses, only 16 licenses are installed.`	More licenses are being used than are installed.	1. Verify the license usage. **show license usage** *license_name* See Example 4-1 on page 4-5. 2. Contact your Cisco Customer Support Account Team to acquire additional licenses.

VEMs fails to acquire licenses even though the **show license usage** command shows there are enough licenses available. The following syslog messages are seen:
`2014 Jun 7 20:15:36 vsm-demo LICMGR-3-LOG_LIC_CHECKOUT_FAIL_BAD_CLOCK: License checkout failed for feature NEXUS1000V_LAN_SERVICES_PKG(VEM 3 - Socket 1(1.0)) because system clock has been set back. Please set the clock to the correct value.`
`2014 Jun 7 20:15:36 vsm-demo VEM_MGR-2-VEM_MGR_UNLICENSED: License for VEM 3 could not be obtained. Please contact your Cisco account team or partner to purchase Licenses or downgrade to Essential Edition. To activate your purchased licenses, click on www.cisco.com/go/license.`
	The clock has been changed back manually or through NTP, which has invalidated evaluation licenses. The problem is seen even if there are enough permanent licenses available to license the VEMs as long as evaluation licenses are present. You can look for the following syslog message to find the time when the clock changed:
	`2014 Jun 7 20:15:24 vsm-demo VEM_MGR-5-VEM_MGR_CLOCK_CHANGE: Clock setting has been changed on the system. Please be aware that, in Advanced edition, clock changes will force a recheckout of all existing VEM licenses. During this recheckout procedure, licensed VEMs which are offline will lose their licenses.`
		1. Undo the clock change using the **clock set** command or uninstall all evaluation licenses using the **clear license** command.
		2. Ensure there are enough permanent licenses available before uninstalling evaluation licenses.
		3. Verify that the modules are licensed using the **show module vem license-info** command.

License Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to licenses.

Command	Purpose
show module vem license-info	Displays the VEM license information including the license type, license status, license version, and socket count. See Example 4-8 on page 4-7.
show license usage [*license_name*]	Displays information about the licenses and where they are used. If displayed for a specific license, indicates VEM and socket information. See Example 4-1 on page 4-5.

Command	Purpose
show interface veth	Displays the messages logged about port profile events within the Cisco Nexus 1000V. See Example 4-2 on page 4-6.
show license host-id	Displays the serial number for your Cisco Nexus 1000V license. See Example 4-3 on page 4-6.
show license file	Displays the contents of a named license file. See Example 4-4 on page 4-6.
svs license transfer src-vem *vem no* **license_pool**	Transfers the licenses from a VEM to the license pool. See Example 4-5 on page 4-6.
show license brief	Displays the version and license count information for each license file. See Example 4-6 on page 4-6.
show switch edition	Displays the switch edition, advanced feature status, license expiry, module and virtual Ethernet scale. Example 4-7 on page 4-7.

For detailed information about **show** command output, see the *Cisco Nexus 1000V Command Reference*.

EXAMPLES

Example 4-1 show license usage *license_name* ***Command***

```
switch# show license usage NEXUS1000V_LAN_SERVICES_PKG
--------------------------------------
Feature Usage Info
--------------------------------------
       Installed Licenses :    10
            Eval Licenses :   0
   Max Overdraft Licenses :    16
Installed Licenses in Use :    4
Overdraft Licenses in Use :    0
     Eval Licenses in Use :    0
       Licenses Available :   22
--------------------------------------
Application
--------------------------------------
VEM 3 - Socket 1
VEM 3 - Socket 2
VEM 4 - Socket 1
VEM 4 - Socket 2
--------------------------------------
switch#
```

Example 4-2 show interface vethernet Command

```
switch# show int veth1
Vethernet1 is down (VEM Unlicensed)
     Port description is VM-Pri, Network Adapter 1
     Hardware is Virtual, address is 0050.56b7.1c7b
     Owner is VM "VM-Pri", adapter is Network Adapter 1
     Active on module 5
     VMware DVS port 32
     Port-Profile is dhcp-profile
     Port mode is access
     Rx
     5002 Input Packets 4008 Unicast Packets
     85 Multicast Packets 909 Broadcast Packets
     846478 Bytes
     Tx
     608046 Output Packets 17129 Unicast Packets
     502543 Multicast Packets 88374 Broadcast Packets 0 Flood Packets
     38144480 Bytes
     20 Input Packet Drops 0 Output Packet Drops
```

Example 4-3 show license host-id Command

```
switch# show license host-id
License hostid: VDH=8449368321243879080
switch#
```

Example 4-4 show license file Command

```
switch# show license file sample.lic
sample.lic:
        SERVER this_host ANY
        VENDOR cisco
        INCREMENT NEXUS1000V_LAN_SERVICES_PKG cisco 3.0 permanent 16 \
        HOSTID=VDH=8449368321243879080 \
        NOTICE="<LicFileID>sample.lic</LicFileID><LicLineID>0</LicLineID> \
        <PAK>dummyPak</PAK>" SIGN=34FCB2B24AE8

switch#
```

Example 4-5 svs license transfer src-vem *vem no* license_pool Command

```
switch# svs license transfer src-vem 3 license_pool
switch#
```

Example 4-6 show license brief Command

```
switch# show license brief
NOTE: * is UPGRADE FILE
-------------------------------------------------------------------------------------------
File Name Feature Name Version Count Expiry
-------------------------------------------------------------------------------------------
eval.lic NEXUS1000V_LAN_SERVICES_PKG 1.0 17 3-nov-2014
eval0715.lic NEXUS1000V_LAN_SERVICES_PKG 3.0 17 15-jul-2015

show switch edition (purpose: Displays the switch edition, advanced feature status,
license expiry and module and veth scale)
```

Example 4-7 show switch edition Command

```
switch# show switch edition
Switch Edition: ADVANCED (3.0)

Feature Status
Name State Licensed In version
--------------------------------------------------------
cts enabled Y 1.0
dhcp-snooping disabled Y 1.0
vxlan-gateway enabled Y 1.0
bgp disabled Y 3.0
bpduguard disabled Y 3.0

License Status
Edition Available In Use Expiry Date
----------------------------------------------
Advanced 17 0 03 Nov 2014

Scale Support
Edition Modules Virtual Ports
---------------------------------------
Advanced 256 12288
```

Example 4-8 show module vem license-info Command

```
n100v# show module vem license-info
Licenses are Sticky
Mod Socket Count License Usage Count License Version License Status
--- ------------ ------------------- --------------- --------------
103 2 2 3.0 licensed
104 2 2 3.0 licensed
```

CHAPTER 5

Upgrades

This chapter describes how to identify and resolve problems related to upgrading the Virtual Supervisor Module (VSM) software and includes the following sections:

Information About Upgrades

The upgrade for the Cisco Nexus 1000V involves upgrading software on both the VSM and the Virtual Ethernet Module (VEM).

An in service software upgrade (ISSU) is available for a stateful upgrade of the Cisco Nexus 1000V image(s) running on the VSM. A stateful upgrade is one without noticeable interruption of data plane services provided by the switch.

For detailed information, see the *Cisco Nexus 1000V Installation and Upgrade Guide*.

Problems with the In-Service Software Upgrade

The following are symptoms, possible causes, and solutions for problems with ISSUs.

Table 5-1 Problems with the ISSU

Symptom	Possible Causes	Solution

Error Message:
`Pre-Upgrade check failed.`
`Return code 0x40930062 (free space in the filesystem is below threshold).`
	This error indicates that there is not enough space in the /var/sysmgr partition.	1. Reboot the system.

Table 5-1 (continued)Problems with the ISSU

Symptom	Possible Causes	Solution
Error message: `Pre-Upgrade check failed. Return code 0x4093000A (SRG collection failed)`	A module is removed during the upgrade.	1. Make sure that the module removal is complete. 2. Restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.
Error message: `Pre-Upgrade check failed. Return code 0x40930076 (Standby sup is offline. ISSU will not proceed)`	The standby VSM is not present or is not synchronized with the active VSM, and the VSMs do not form a stable HA pair.	1. Verify the HA synchronization state. **show system redundancy status** The output of the **show** command must indicate the following: Active VSM: Active with HA standby Standby VSM: HA standby 2. If the output of the **show** command indicates that the VSMs are not synchronized, see the "Problems with High Availability" section on page 6-2. 3. When the VSMs are synchronized, restart the software upgrade using the detailed instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.

Error message:
`Pre-Upgrade check failed. Return code 0x807B0002 (No such file or directory)`
Error message:
`Pre-Upgrade check failed. Return code 0x4093000F (Failed to copy image)`
	The software image files required for the upgrade are not present or were not copied to the bootflash: repository.
	There may not be enough room in the bootflash: repository for the files to be copied.
		1. Verify that there is enough space in bootflash for the image files.
		dir
		2. Do one of the following:
		– If additional space is needed, delete other files from the bootflash: repository to make room for the software image files.
		delete
		Caution Do not delete kickstart or system image files from bootflash. If there are no image files in bootflash, the system cannot reboot if required.
		– If not, continue with the next step.
		3. Download the required images from www.cisco.com to the bootflash: repository.
		4. Verify that the correct images are in the bootflash: repository.
		show boot
		5. When the correct software images are in the bootflash: repository, restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.

Table 5-1 (continued)Problems with the ISSU

Symptom	Possible Causes	Solution

The **install** command fails with the following error:
`Return code 0x4045001F (image MD5 checksum error)`
`Pre-Upgrade check failed. Return code 0x40930011 (Image verification failed)`
	The software image file(s) required for the upgrade do not pass the MD5 checksum verification, indicating that the correct file(s) are not present in bootflash: repository for the upgrade to proceed.
	A file can be truncated when copied.
		1. Using the README file from the upgrade zip folder at www.cisco.com, verify the MD5 checksum for each of the image files.
		show file bootflash: *filename* **md5sum**
		2. Replace the file(s) that do not match.
		3. Verify that the correct images are in the bootflash: repository and that the checksums match.
		show file bootflash: *filename* **md5sum**
		4. When the correct software images are in the bootflash: repository, restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.

Error message: `Install has failed. Return code 0x40970001 (Incompatible image)`	You might have used an incorrect filename when entering the in**stall all** command.	Restart the software upgrade using the correct filenames for the new software images. **`install all kickstart`** *filename1* **`system`** *filename2*
After upgrading, the VSMs are not running the new software version.	The boot variables were not set properly.	1. Verify that the running images and boot variables match the upgrade version. **show version** **show boot** 2. If needed, download the required images from www.cisco.com to your local bootflash: repository. 3. Verify that the correct images are in the bootflash: repository. **show boot** 4. Restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*. 5. If the problem persists, collect details of the upgrade and open a support case. **show system internal log install details**

Performing the configuration copy process fails and stops the upgrade.
`Performing configuration copy.`
`[####------------] 30%`
	Service or system errors.	1. Manually copy the configuration.
		copy running-config startup-config
		2. Do one of the following:
		– If the progress bar gets stuck before 100% for over one minute, collect details of the upgrade and open a support case.
		show system internal log install details
		– If the copy succeeds without delays, restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.

Table 5-1 (continued)Problems with the ISSU

Symptom	Possible Causes	Solution
Error message: `Another install procedure may be in progress. (0x401E0007)`	Another upgrade session is in progress from a VSM console or SSH/Telnet.	Do one of the following: • Continue the first upgrade session in progress. • Stop the upgrade and restart one session only using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.
The **install** command fails with following error message: `-- FAIL. Return code 0x4093001E (Standby failed to come online) Install has failed. Return code 0x4093001E (Standby failed to come online)`	The standby VSM fails to boot with the new image.	Do one of the following: • Restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*. • Postpone the upgrade and reset the boot variables to the original filenames. **boot kickstart** *filename* [**sup-1**] [**sup-2**]
The **install** command fails with following error message: `Install has failed. Return code 0x4093001F (Standby installer failed to take over the installation). Please identify the cause of the failure, and try "install all" again"`	The standby VSM takes more than 10 minutes to come up and form a stable HA pair with the active VSM.	1. Reset the boot variables to the original filenames. **boot kickstart** *filename* [**sup-1**] [**sup-2**] 2. If the standby is still running the new software version, reload it. **reload** The standby synchronizes with the active, so that both are running the original software version.
The **install** command fails with following error message: `Module 2: Waiting for module online. -- SUCCESS -- Install has failed. Return code 0x40930000 (Current operation failed to complete within specified time)`	A failure at the standby VSM caused it to reload again after the **Continuing with installation, please wait** message and before the switchover.	1. Inspect the logs. **show logging** 2. Look for standby reloads caused by process failures. **show cores** If a process crash is observed, collect details of the upgrade and open a support case. **show system internal log install details** 3. Restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.
The pre-upgrade check failed: Return code 0x40930062 (free space in the filesystem is below threshold).		1.

Problems with the VEM Upgrade

The following are symptoms, possible causes, and solutions for problems with VEM software upgrade.

Table 5-2 Problems with the VEM Upgrade

Symptom	Possible Causes	Solution
After starting a VEM upgrade from the VSM console, the VMware Upgrade Manager (VUM) skips upgrading the hosts with the new VEM.	One or more of the following are enabled on the host cluster. • VMware high availability (HA) • VMware fault tolerance (FT) • Vmware Distributed Power Management (DPM)	1. Verify the upgrade failure. **show vmware vem upgrade status** 2. From vCenter Server, disable HA, FT, and DPM for the cluster. 3. Restart the VEM software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.
VEM upgrade fails.	An incorrect VUM version is in use.	1. Identify the VUM version required for the upgrade using the *Cisco Nexus 1000V Compatibility Information*. 2. Upgrade to the correct VUM version. 3. Restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.
After upgrading, the host is not added to the VSM.	An incorrect VEM software version is installed on the host.	1. Identify the VEM software version required for the upgrade using the *Cisco Nexus 1000V Compatibility Information*. 2. Proceed with the upgrade using the correct VEM software version and the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.
A message on the ESX/ESXi command line shell and VMkernel logs notifies you that the loading and unloading of modules failed.	The modules were not placed in maintenance mode (all VMs VMotioned over) before starting the upgrade.	1. Place the host in maintenance mode. 2. Proceed with the upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.
	The host does not have enough memory to load new modules. A host requires a minimum of 2 GB of physical RAM. If it also hosts a Cisco Nexus 1000V VSM VM, it needs a minimum of 4 GB of physical RAM. If it also hosts the vCenter Server VM, additional memory might be needed.	1. Verify that the host has sufficient memory to load the new modules. For more information about allocating RAM and CPU, see the *Cisco Nexus 1000V Installation and Upgrade Guide*. 2. Proceed with the upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.

Problems with the GUI Upgrade

The following are symptoms, possible causes, and solutions for problems with software upgrade using the GUI upgrade application.

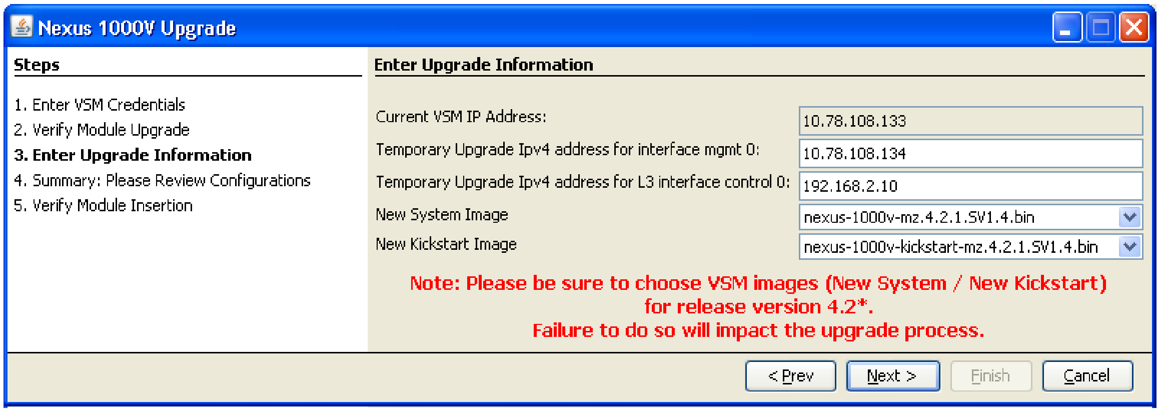
Note If you are upgrading directly from SV1(4) to SV1(4a), the GUI is not used and this section does not apply. This section is applicable only if you use the GUI for an intermediate upgrade from a SV1(3x) release to SV1(4), prior to upgrading to SV1(4a).

Table 5-3 Problems with the GUI Upgrade

Symptom	Possible Causes	Solution
The upgrade GUI stops and times out after 10 minutes and displays the following message: `Error: Could not contact the upgraded VSM at` *`n.n.n.n`*`. Please check the connection.`	During the upgrade, you configured an unreachable IP address for the mgmt0 interface. In this case, one VSM in the redundant pair has new software installed and is unreachable. The other VSM has the original pre-upgrade software version installed and is reachable.	**1.** Use one of the following sets of procedures to return your VSM pair to the previous software version: – "Recovering a Secondary VSM with Active Primary" section on page 5-7 – "Recovering a Primary VSM with Active Secondary" section on page 5-12 **2.** Restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.
The upgrade GUI stops and times out after 10 minutes and displays the following message: `Error: Could not contact the upgraded VSM at 10.104.244.150. Please check the connection.` After timing out, one VSM comes up in switch(boot) mode.	You have selected incompatible or incorrect VSM software images for the upgrade. The software images you selected from the GUI selection list included a system image for one software version and a kickstart image for another software version. These images must be for the same software version. For an example of how software images are selected during the upgrade, see Example 5-1.	**1.** To continue the upgrade, first recover the VSM using one of the following: – "Recovering a Secondary VSM with Active Primary" section on page 5-7 – "Recovering a Primary VSM with Active Secondary" section on page 5-12 **2.** Restart the software upgrade using the instructions in the *Cisco Nexus 1000V Installation and Upgrade Guide*.

Example 5-1 Upgrade: Enter Upgrade Information

This example shows how to specify system and kickstart images during the upgrade process. In this example, the images specified are from the same release, SV1.4. If you specify a kickstart image from one release, and a system image from another, then the upgrade cannot proceed.



Steps

1. Enter VSM Credentials
2. Verify Module Upgrade
3. **Enter Upgrade Information**
4. Summary: Please Review Configurations
5. Verify Module Insertion

Enter Upgrade Information

Current VSM IP Address:	10.78.108.133
Temporary Upgrade Ipv4 address for interface mgmt 0:	10.78.108.134
Temporary Upgrade Ipv4 address for L3 interface control 0:	192.168.2.10
New System Image	nexus-1000v-mz.4.2.1.SV1.4.bin
New Kickstart Image	nexus-1000v-kickstart-mz.4.2.1.SV1.4.bin

Note: Please be sure to choose VSM images (New System / New Kickstart) for release version 4.2*.
Failure to do so will impact the upgrade process.

Recovering a Secondary VSM with Active Primary

You can recover a secondary VSM when the primary VSM is active.

Note The information in this section does not apply when upgrading from Release 4.2(1)SV1(4) to Release 4.2(1)SV2(1.1).

Step 1 Stop the upgrade on the VSM, using the "Stopping a VSM Upgrade" section on page 5-8

Step 2 Change the boot variables back to the previous version using the "Changing Boot Variables" section on page 5-9

Step 3 From the vCenter Server left-hand panel, right-click the secondary VSM and then choose **Delete from Disk**.

The secondary VSM is deleted.

Step 4 Create a new VSM by reinstalling the software using the vSphere Client Deploy OVF Template wizard, specifying the following:

- The Cisco Nexus 1000V secondary configuration method
(configures the secondary VSM in an HA pair using a GUI setup dialog).
- The host or cluster of the primary VSM.
- The same domain ID and password as that of the primary VSM.

For a detailed procedure, see the *Cisco Nexus 1000V Installation and Upgrade Guide*.

The VSM comes up and forms an HA pair with the newly created standalone VSM. The VSMs have the previous version of the software installed.

Stopping a VSM Upgrade

You can stop a VSM upgrade that is in progress.

BEFORE YOU BEGIN

- Log in to the CLI in EXEC mode.

Note The information in this section does not apply when upgrading from Release 4.2(1)SV1(4) to Release 4.2(1)SV2(1.1).

DETAILED STEPS

Step 1 Display upgrade status.

show svs upgrade status

```
Example:
switch# show svs upgrade status
Upgrade State: Start
Upgrade mgmt0 ipv4 addr: 1.1.1.1
Upgrade mgmt0 ipv6 addr:
Upgrade control0 ipv4 addr:
```

Step 2 Stop the upgrade.

a. **configure terminal**

b. **no svs upgrade start**

```
Example:
switch# configure terminal
switch#(config)# no svs upgrade start
WARNING! VSM upgrade process is aborted
switch#(config)#
```

Step 3 Display the upgrade status.

show svs upgrade status

```
Example:
switch#(config)# show svs upgrade status
Upgrade State: Abort
Upgrade mgmt0 ipv4 addr:
Upgrade mgmt0 ipv6 addr:
Upgrade control0 ipv4 addr:
```

Step 4 You have completed this procedure. Return to one of these sections:

- "Recovering a Secondary VSM with Active Primary" section on page 5-7
- "Recovering a Primary VSM with Active Secondary" section on page 5-12

Changing Boot Variables

You can replace the software images used to boot the VSM.

BEFORE YOU BEGIN

- Log in to the CLI in EXEC mode.
- Know the filenames of the pre-upgrade system and kickstart image files to apply.

DETAILED STEPS

Step 1 Display the current boot variables.

show boot

```
Example:
switch# show boot
sup-1
kickstart variable = bootflash:/nexus-1000v-kickstart-mz.4.0.4.SV1.3a.bin
system variable = bootflash:/nexus-1000v-mz.4.0.4.SV1.3a.bin
sup-2
kickstart variable = bootflash:/nexus-1000v-kickstart-mzg.4.2.1.SV1.4.bin
system variable = bootflash:/nexus-1000v-mzg.4.2.1.SV1.4.bin
No module boot variable set
switch(config)#
```

Step 2 Remove the current system and kickstart boot variables.

a. **configure terminal**

b. **no boot system**

c. **no boot kickstart**

```
Example:
switch# configure terminal
switch(config)# no boot system
switch(config)# no boot kickstart
switch#(config)#
```

Step 3 Restore the system and kickstart boot variables to the original pre-upgrade filenames.

a. **boot system bootflash:***system-boot-variable-name*

b. **boot system bootflash:***kickstart-boot-variable-name*

```
Example:
switch#(config)# boot system bootflash:nexus-1000v-mz.4.0.4.SV1.3a.bin
switch#(config)# boot kickstart bootflash:nexus-1000v-kickstart-mz.4.0.4.SV1.3a.bin
switch#(config)#
```

Step 4 Copy the running configuration to the startup configuration.

copy run start

```
Example:
switch#(config)# copy run start
[########################################] 100%e
switch#(config)#
```

Step 5 Verify the change in the system and kickstart boot variables.

show boot

```
Example:
switch#(config)# show boot
sup-1
kickstart variable = bootflash:/nexus-1000v-kickstart-mz.4.0.4.SV1.3a.bin
system variable = bootflash:/nexus-1000v-mz.4.0.4.SV1.3a.bin
sup-2
kickstart variable = bootflash:/nexus-1000v-kickstart-mz.4.0.4.SV1.3a.bin
system variable = bootflash:/nexus-1000v-mz.4.0.4.SV1.3a.bin
No module boot variable set
switch#(config)#
```

Step 6 You have completed this procedure. Return to one of these sections:

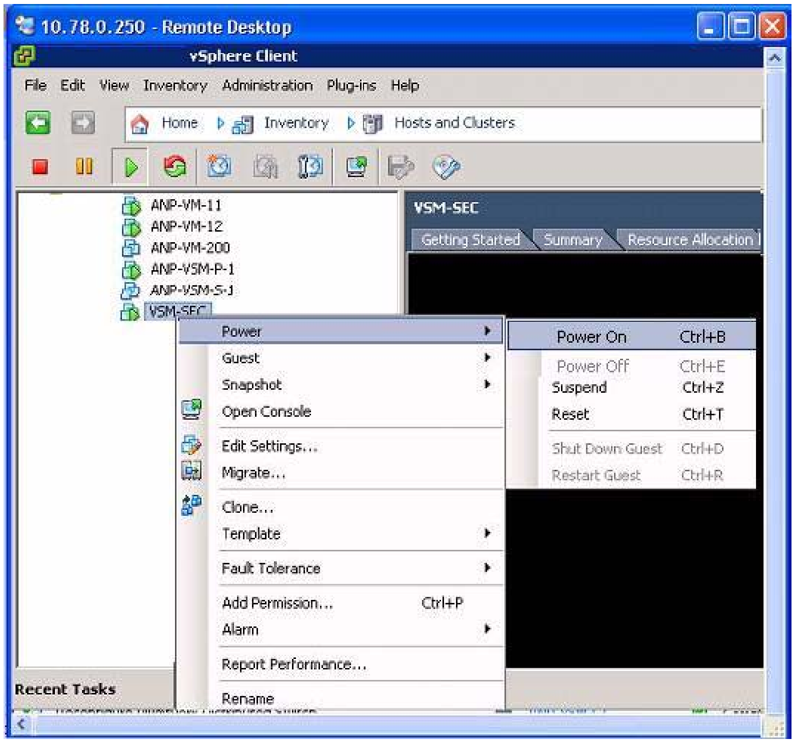
- "Recovering a Secondary VSM with Active Primary" section on page 5-7
- "Recovering a Primary VSM with Active Secondary" section on page 5-12

Powering On the VSM

You can power on the newly created VSM.

Step 1 From the vCenter Server left-hand panel, right-click the VSM and then choose **Power > Power On**.

The VSM starts.



Step 2 You have completed this procedure. Return to the "Recovering a Primary VSM with Active Secondary" section on page 5-12.

Changing the HA Role

You can change the HA role of the VSM.

BEFORE YOU BEGIN

- Log in to the CLI in EXEC mode.
- Know the domain ID of the existing VSM.

DETAILED STEPS

Step 1 Go to the domain of the existing VSM.

a. **configure terminal**

b. **svs-domain**

c. **domain id** *domain-id*

Example:

```
switch# config t
switch(config)# svs-domain
switch(config-svs-domain)# domain id 1941
Warning: Config saved but not pushed to vCenter Server due to inactive connection!
```

Step 2 Change the HA role.

system redundancy role [primary | secondary | standalone]

Example:

```
switch(config-svs-domain)# system redundancy role secondary
Setting will be activated on next reload.
switch(config-svs-domain)#
```

Example:

```
switch(config-svs-domain)# system redundancy role primary
Setting will be activated on next reload.
switch(config-svs-domain)#
```

Step 3 Copy the running configuration to the startup configuration.

copy run start

Example:

```
switch#(config-svs-domain)# copy run start
[########################################] 100%e
switch#(config-svs-domain)#
```

Step 4 You have completed this procedure. Return to the "Recovering a Primary VSM with Active Secondary" section on page 5-12.

Recovering a Primary VSM with Active Secondary

You can recover a primary VSM when the secondary VSM is active.

Step 1 Stop the upgrade on the secondary VSM by completing the "Stopping a VSM Upgrade" procedure on page 5-8

Step 2 Change the boot variables back to the previous version by completing the "Changing Boot Variables" procedure on page 5-9

Step 3 From the vCenter Server left-hand panel, right-click the primary VSM and then choose **Delete from Disk**.

The primary VSM is deleted.

Step 4 Create a new VSM by reinstalling the software from the OVA and specifying the following:

- Manual (CLI) configuration method instead of GUI.
- The host or cluster of the existing secondary VSM.

For detailed installation procedures, see the *Cisco Nexus 1000V Installation and Upgrade Guide*.

Step 5 Make sure that the port groups between the host server and VSM are not connected when the new VSM is powered on by completing the "Disconnecting the Port Groups" procedure on page 5-12.

Step 6 Power on the newly-created VSM by completing the "Powering On the VSM" procedure on page 5-10.

The VSM comes up with the standalone HA role.

Step 7 Change the HA role of the newly created standalone VSM to primary and save the configuration by completing the "Changing the HA Role" procedure on page 5-11.

Step 8 Power off the newly created VSM by completing the "Powering Off the VSM" procedure on page 5-14.

Step 9 Make sure that the port groups between the host server and VSM are connected when the new VSM is powered on by completing the "Connecting the Port Groups" procedure on page 5-14.

Step 10 Power on the newly created VSM by completing the "Powering On the VSM" procedure on page 5-10.

The VSM comes up, connects with the host server, and forms an HA pair with the existing primary VSM.

Disconnecting the Port Groups

You can disconnect and prevent port groups to the VSM from connecting to the host server.

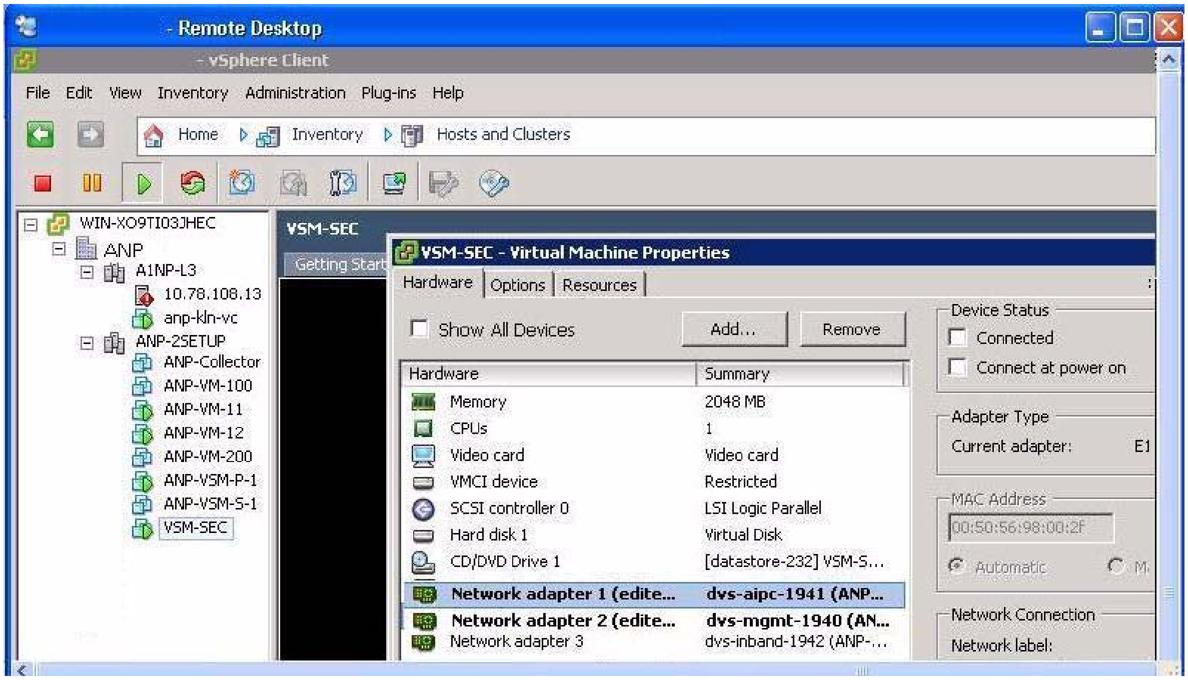
Step 1 In vCenter Server, select the VSM and then choose **Edit > Settings**.

The Virtual Machine Properties dialog box opens.

Step 2 Select the Control port group and uncheck the following Device Settings:

- Connected
- Connect at Power On

The connection from the VSM to the host server through the control port is dropped and is not restored when you power on the VSM.



□ Show All Devices

Hardware	Summary
Memory	2048 MB
CPUs	1
Video card	Video card
VMCI device	Restricted
SCSI controller 0	LSI Logic Parallel
Hard disk 1	Virtual Disk
CD/DVD Drive 1	[datastore-232] VSM-S...
Network adapter 1 (edite...	**dvs-aipc-1941 (ANP...**
Network adapter 2 (edite...	**dvs-mgmt-1940 (AN...**
Network adapter 3	dvs-inband-1942 (ANP-...

Device Status
□ Connected
□ Connect at power on

Adapter Type
Current adapter: E1

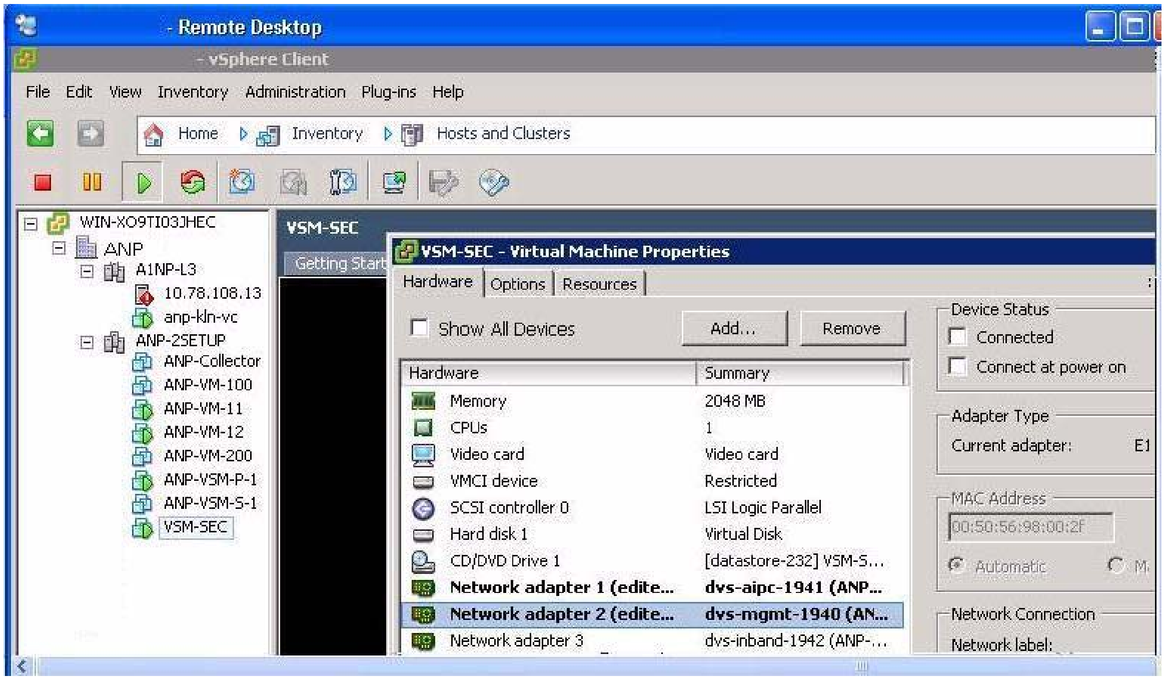
MAC Address
00:50:56:98:00:2f
● Automatic ○ M

Network Connection
Network label:

Step 3 Select the Management port group and uncheck the following Device Settings:

- Connected
- Connect at Power On

The connection from the VSM to the host server through the management port is dropped and is not restored when you power on the VSM.



□ Show All Devices

Hardware	Summary
Memory	2048 MB
CPUs	1
Video card	Video card
VMCI device	Restricted
SCSI controller 0	LSI Logic Parallel
Hard disk 1	Virtual Disk
CD/DVD Drive 1	[datastore-232] VSM-S...
Network adapter 1 (edite...	**dvs-aipc-1941 (ANP...**
Network adapter 2 (edite...	**dvs-mgmt-1940 (AN...**
Network adapter 3	dvs-inband-1942 (ANP-...

Device Status
□ Connected
□ Connect at power on

Adapter Type
Current adapter: E1

MAC Address
00:50:56:98:00:2f
● Automatic ○ M

Network Connection
Network label:

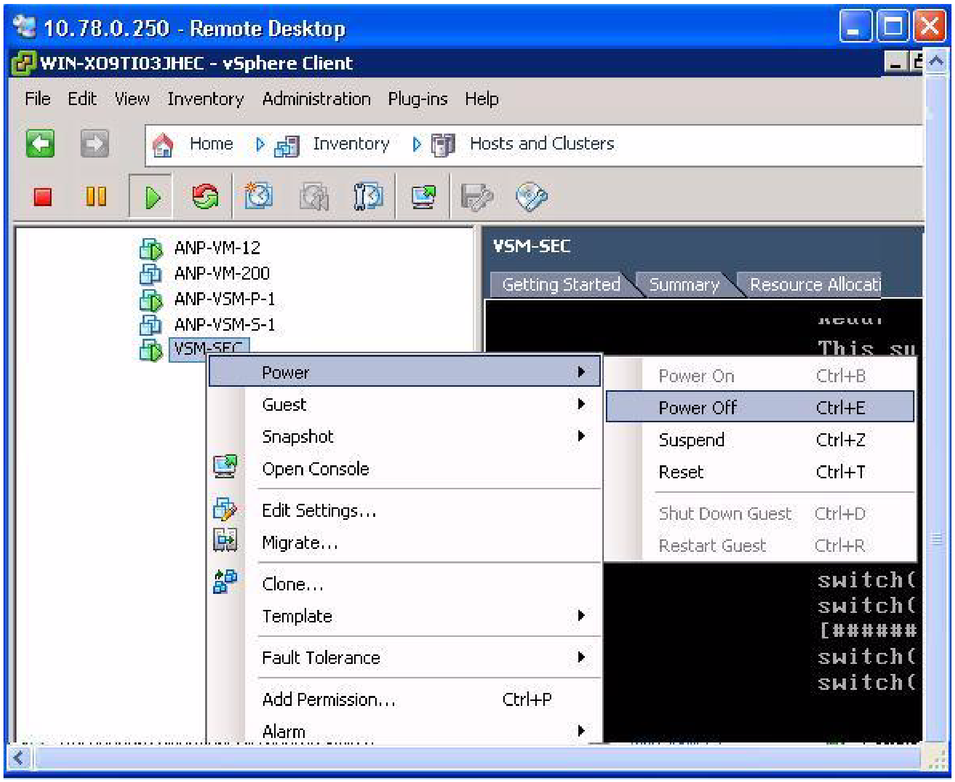
Step 4 You have completed this procedure. Return to the "Recovering a Primary VSM with Active Secondary" section on page 5-12.

Powering Off the VSM

You can power off the newly created VSM.

Step 1 From the vCenter Server left-hand panel, right-click the VSM and then choose **Power > Power Off**.

The VSM shuts down.



Step 2 You have completed this procedure. Return to the "Recovering a Primary VSM with Active Secondary" section on page 5-12.

Connecting the Port Groups

You can make sure that the port groups to the host connect when you power on the VSM.

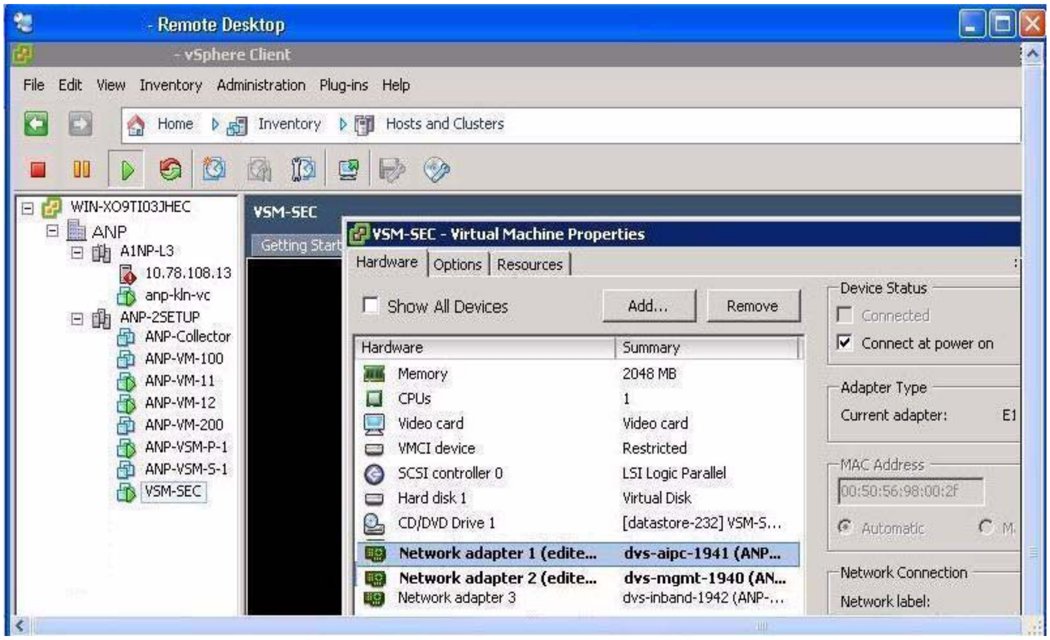
Step 1 In vCenter Server, select the VSM and then choose **Edit > Settings**.

The Virtual Machine Properties dialog box opens.

Step 2 Select the Control port group and check the following Device Settings:

- Connect at Power On

When you power on the VSM, it will connect to the host server through the control port.



□ Show All Devices

Hardware	Summary
Memory	2048 MB
CPUs	1
Video card	Video card
VMCI device	Restricted
SCSI controller 0	LSI Logic Parallel
Hard disk 1	Virtual Disk
CD/DVD Drive 1	[datastore-232] VSM-S...
Network adapter 1 (edite...	**dvs-aipc-1941 (ANP...**
Network adapter 2 (edite...	**dvs-mgmt-1940 (AN...**
Network adapter 3	dvs-inband-1942 (ANP-...

Device Status
□ Connected
☑ Connect at power on

Adapter Type
Current adapter: E1

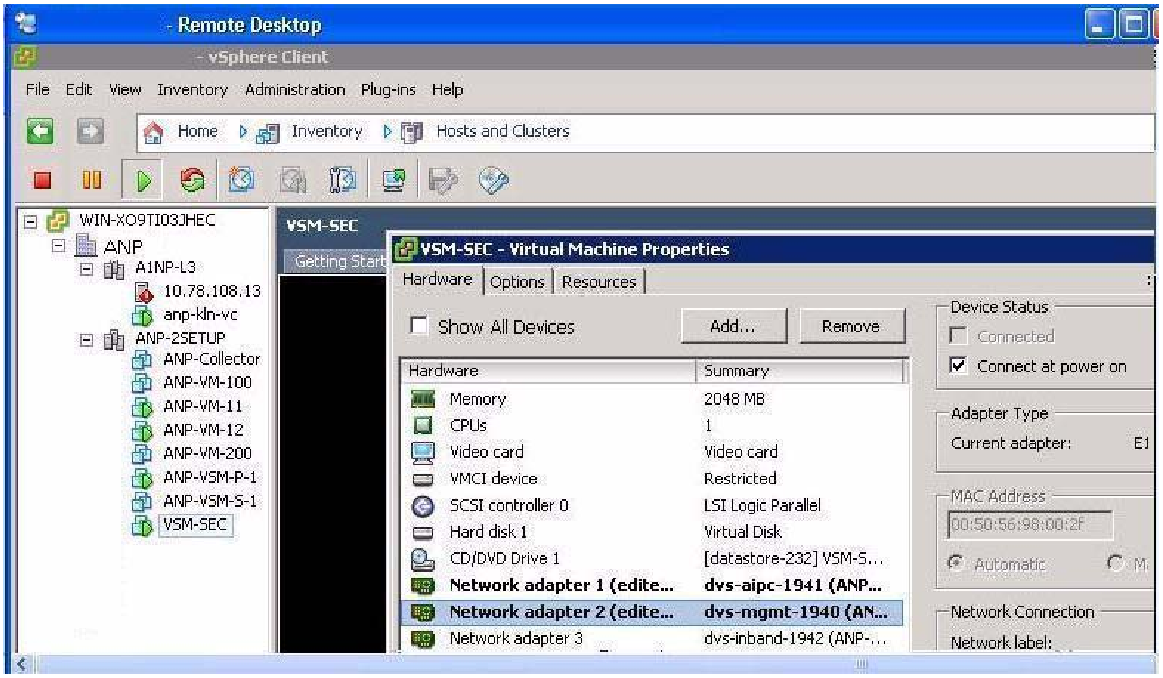
MAC Address
00:50:56:98:00:2f
● Automatic ○ M

Network Connection
Network label:

Step 3 Select the Management port group and check the following Device Setting:

- Connect at Power On

When you power on the VSM, it will connect to the host server through the management port.



□ Show All Devices

Hardware	Summary
Memory	2048 MB
CPUs	1
Video card	Video card
VMCI device	Restricted
SCSI controller 0	LSI Logic Parallel
Hard disk 1	Virtual Disk
CD/DVD Drive 1	[datastore-232] VSM-S...
Network adapter 1 (edite...	**dvs-aipc-1941 (ANP...**
Network adapter 2 (edite...	**dvs-mgmt-1940 (AN...**
Network adapter 3	dvs-inband-1942 (ANP-...

Device Status
□ Connected
☑ Connect at power on

Adapter Type
Current adapter: E1

MAC Address
00:50:56:98:00:2f
● Automatic ○ M

Network Connection
Network label:

Step 4 You have completed this procedure. Return to the "Recovering a Primary VSM with Active Secondary" section on page 5-12.

Problems with VSM-VEM Layer 2 to 3 Conversion Tool

The following is a symptom and solution for a problem with logging in to VSM when using the conversion tool:

Symptom	Solution
When you enter your VSM and VC login credentials for the first time, the VSM-VEM Layer 2 to 3 Conversion Tool might display: `Timeout error. Is device down or unreachable?? ssh_expect`	1. Open a command line window and run an **ssh** command on the VSM (**ssh** *username@vsmIPaddress*). 2. When prompted, `Are you sure you want to continue connecting?`, enter **yes**. 3. Rerun the VSM-VEM Layer 2 to 3 Conversion Tool by reopening the .bat file. Ensure that the error does not reappear.

Upgrade Troubleshooting Commands

You can troubleshoot problems related to upgrades.

Command	Description
show boot	Displays boot variable definitions, showing the names of software images used to boot the VSM. See Example 5-2 on page 5-17.
show module	Displays module status for active and standby VSMs. See Example 5-4 on page 5-17 (ISSU). See Example 5-4 on page 5-17.
show running-config | include boot	Displays the boot variables currently in the running configuration. See Example 5-5 on page 5-18.
show startup-config | include boot	Displays the boot variables currently in the startup configuration. See Example 5-6 on page 5-18.
show svs connections	Displays the current connections between the VSM and the VMware host server. See Example 5-7 on page 5-18.

Command	Description
show svs upgrade status	Displays the upgrade status. See Example 5-8 on page 5-18.
show system redundancy status	Displays the current redundancy status for the VSM. See Example 5-9 on page 5-19.
show vmware vem upgrade status	Displays the upgrade status. See Example 5-10 on page 5-19.

Example 5-2 show boot Command

```
switch# show boot
sup-1
kickstart variable = bootflash:/nexus-1000v-kickstart-mz.4.0.4.SV1.3a.bin
system variable = bootflash:/nexus-1000v-mz.4.0.4.SV1.3a.bin
sup-2
kickstart variable = bootflash:/nexus-1000v-kickstart-mzg.4.2.1.SV1.4.bin
system variable = bootflash:/nexus-1000v-mzg.4.2.1.SV1.4.bin
No module boot variable set
switch#
```

Example 5-3 show module Command (VSM upgraded first with ISSU, VEM upgrade pending)

```
switch# show module
Mod Ports Module-Type                      Model              Status
--- ----- -------------------------------- ------------------ ------------
1   0     Virtual Supervisor Module        Nexus1000V         ha-standby
2   0     Virtual Supervisor Module        Nexus1000V         active *
3   248   Virtual Ethernet Module          NA                 ok
Mod Sw              Hw
--- --------------- ------
1   4.2(1)SV1(4a)   0.0
2   4.2(1)SV1(4a)   0.0
3   4.2(1)SV1(4)    1.9
Mod MAC-Address(es)                        Serial-Num
--- -------------------------------------- ----------
1   00-19-07-6c-5a-a8 to 00-19-07-6c-62-a8 NA
2   00-19-07-6c-5a-a8 to 00-19-07-6c-62-a8 NA
3   02-00-0c-00-03-00 to 02-00-0c-00-03-80 NA
Mod Server-IP       Server-UUID                          Server-Name
--- --------------- ------------------------------------ --------------------
1   10.78.109.43    NA                                   NA
2   10.78.109.43    NA                                   NA
3   10.78.109.51    4220900d-76d3-89c5-17d7-b5a7d1a2487f 10.78.109.51
switch#
```

Example 5-4 show module Command (VEM and VSM upgraded)

```
switch# show module
Mod  Ports  Module-Type                      Model              Status
---  -----  -------------------------------- ------------------ ------------
1    0      Virtual Supervisor Module        Nexus1000V         ha-standby
2    0      Virtual Supervisor Module        Nexus1000V         active *
3    248    Virtual Ethernet Module          NA                 ok

Mod  Sw               Hw
---  ---------------  ------
```

```
1    4.0(4)SV1(3)     0.0
2    4.0(4)SV1(3)     0.0
3    4.2(1)SV1(4)     1.9

Mod  MAC-Address(es)                         Serial-Num
---  --------------------------------------  ----------
1    00-19-07-6c-5a-a8 to 00-19-07-6c-62-a8  NA
2    00-19-07-6c-5a-a8 to 00-19-07-6c-62-a8  NA
3    02-00-0c-00-03-00 to 02-00-0c-00-03-80  NA

Mod  Server-IP        Server-UUID                           Server-Name
---  ---------------  ------------------------------------  --------------------
1    10.78.109.43     NA                                    NA
2    10.78.109.43     NA                                    NA
3    10.78.109.51     4220900d-76d3-89c5-17d7-b5a7d1a2487f  10.78.109.51
switch#
```

Example 5-5 show running-config | include boot Command

```
switch# show running-config | include boot
boot kickstart bootflash:/nexus-1000v-kickstart-mzg.4.2.1.SV1.4a.bin sup-1
boot system bootflash:/nexus-1000v-mzg.4.2.1.SV1.4a.bin sup-1
boot kickstart bootflash:/nexus-1000v-kickstart-mzg.4.2.1.SV1.4a.bin sup-2
boot system bootflash:/nexus-1000v-mzg.4.2.1.SV1.4a.bin sup-2
switch#
```

Example 5-6 show startup-config | include boot Command

```
switch# show startup-config | include boot
boot kickstart bootflash:/nexus-1000v-kickstart-mzg.4.2.1.SV1.4a.bin sup-1
boot system bootflash:/nexus-1000v-mzg.4.2.1.SV1.4a.bin sup-1
boot kickstart bootflash:/nexus-1000v-kickstart-mzg.4.2.1.SV1.4a.bin sup-2
boot system bootflash:/nexus-1000v-mzg.4.2.1.SV1.4a.bin sup-2
switch#
```

Example 5-7 show svs connections Command

```
switch# show svs connections

connection vc:
    hostname: 172.23.232.139
    remote port: 80
    protocol: vmware-vim https
    certificate: default
    datacenter name: Hamilton-DC
    DVS uuid: 9b dd 36 50 2e 27 27 8b-07 ed 81 89 ef 43 31 17
    config status: Enabled
    operational status: Connected
    sync status: -
    version: -
switch#
```

Example 5-8 show svs upgrade status Command

```
switch# show svs upgrade status
Upgrade State: Start
Upgrade mgmt0 ipv4 addr: 1.1.1.1
Upgrade mgmt0 ipv6 addr:
Upgrade control0 ipv4 addr:
switch#
```

Example 5-9 show system redundancy status Command

```
switch# show system redundancy status
Redundancy role
---------------
      administrative:   primary
         operational:   primary

Redundancy mode
---------------
      administrative:   HA
         operational:   HA

This supervisor (sup-1)
-----------------------
    Redundancy state:   Active
    Supervisor state:   Active
      Internal state:   Active with HA standby

Other supervisor (sup-2)
-----------------------
    Redundancy state:   Standby
    Supervisor state:   HA standby
      Internal state:   HA standby

switch#
```

Example 5-10 show vmware vem upgrade status Command

```
switch# show vmware vem upgrade status
Upgrade VIBs: System VEM Image
Upgrade Status:
Upgrade Notification Sent Time:
Upgrade Status Time(vCenter):
Upgrade Start Time:
Upgrade End Time(vCenter):
Upgrade Error:
Upgrade Bundle ID:
    VSM: VEM400-201007101-BG
    DVS: VEM400-201007101-BG

switch#
```

CHAPTER 6

High Availability

This chapter describes how to identify and resolve problems related to high availability, and includes the following sections:

- Information About High Availability, page 6-1
- Problems with High Availability, page 6-2
- High Availability Troubleshooting Commands, page 6-5

Information About High Availability

The purpose of high availability (HA) is to limit the impact of failures—both hardware and software—within a system. The Cisco NX-OS operating system is designed for high availability at the network, system, and service levels.

The following Cisco NX-OS features minimize or prevent traffic disruption if a failure occurs:

- Redundancy—Redundancy at every aspect of the software architecture.
- Isolation of processes—Isolation between software components to prevent a failure within one process disrupting other processes.
- Restartability—Most system functions and services are isolated so that they can be restarted independently after a failure while other services continue to run. In addition, most system services can perform stateful restarts, which allow the service to resume operations transparently to other services.
- Supervisor stateful switchover—Active/standby dual supervisor configuration. The state and configuration remain constantly synchronized between two Virtual Supervisor Modules (VSMs) to provide a seamless and statefu1 switchover if a VSM failure occurs.

The Cisco Nexus 1000V system is made up of the following:

- Virtual Ethernet Modules (VEMs) running within virtualization servers. These VEMs are represented as modules within the VSM.
- A remote management component, such as VMware vCenter Server.
- One or two VSMs running within virtual machines (VMs).

System-Level High Availability

The Cisco Nexus 1000V supports redundant VSM virtual machines—a primary and a secondary—running as an HA pair. Dual VSMs operate in an active/standby capacity in which only one of the VSMs is active at any given time, while the other acts as a standby backup. The state and configuration remain constantly synchronized between the two VSMs to provide a statefu1 switchover if the active VSM fails.

Network-Level High Availability

The Cisco Nexus 1000V HA at the network level includes port channels and Link Aggregation Control Protocol (LACP). A port channel bundles physical links into a channel group to create a single logical link that provides the aggregate bandwidth of up to eight physical links. If a member port within a port channel fails, the traffic previously carried over the failed link switches to the remaining member ports within the port channel.

Additionally, LACP allows you to configure up to 16 interfaces into a port channel. A maximum of eight interfaces can be active, and a maximum of eight interfaces can be placed in a standby state.

For additional information about port channels and LACP, see the *Cisco Nexus 1000V Layer 2 Switching Configuration Guide*.

Problems with High Availability

Symptom	Possible Causes	Solution
The active VSM does not see the standby VSM.	MAC addresses mismatch. • Check that the peer VSM MAC addresses that are learned by the active VSM by using the **show system redundancy status** command.	Confirm that the standby VSM MAC addresses are correctly learned by the active VSM. 1. Compare the standby VSM MAC addresses with the output MAC addresses by using the **show system redundancy status** command on the active VSM. 2. If the compared MAC addresses are different, use the **peer-sup mac-addresses clear** command to clear the stale MAC addresses that are learned by the active VSM.

Symptom	Possible Causes	Solution
The active VSM does not see the standby VSM.	Roles are not configured properly. • Check the role of the two VSMs by using the **show system redundancy status** command.	1. Confirm that the roles are the primary and secondary role, respectively. 2. If needed, use the **system redundancy role** command to correct the situation. 3. Save the configuration if roles are changed.
	Network connectivity problems. • Check that the control and management VLAN connectivity between the VSM at the upstream and virtual switches.	If network problems exist, do the following: 1. From vSphere Client, shut down the VSM, which should be in standby mode. 2. From vSphere Client, bring up the standby VSM after network connectivity is restored.
The active VSM does not complete synchronization with the standby VSM.	Version mismatch between VSMs. • Check that the primary and secondary VSMs are using the same image version by using the **show version** command.	If the active and the standby VSM software versions differ, reinstall the secondary VSM with the same version used in the primary.
	Fatal errors during gsync process. • Check the gsyncctrl log using the **show system internal log sysmgr gsyncctrl** command and look for fatal errors.	Reload the standby VSM using the **reload module** *module-number* command, where *module-number* is the module number for the standby VSM.
	• The VSM has connectivity only through the management interface. • Check the output of the **show system internal redundancy info** command and verify if the *degraded_mode* flag is set to *true*.	Check control VLAN connectivity between the primary and the secondary VSMs.

Symptom	Possible Causes	Solution
The standby VSM reboots periodically.	The VSM has connectivity only through the management interface. - Check the output of the **show system internal redundancy info** command and verify that the *degraded_mode* flag is set to true.	Check the control VLAN connectivity between the primary and the secondary VSMs.
	The VSMs have different versions. Enter the **debug system internal sysmgr all** command and look for the active_verctrl entry that indicates a version mismatch, as the following output shows: `2009 May  5 08:34:15.721920 sysmgr: active_verctrl: Stdby running diff version- force download the standby sup.`	Isolate the standby VSM and boot it. Use the **show version** command to check the software version in both VSMs. Install the image matching the active VSM on the standby.
Active-Active detected and resolved	When control and management connectivity between the active and the standby goes down for 6 seconds, the standby VSM transitions to the active state. Upon restoration of control and management connectivity, both VSMs detect an active-active condition.	1. Once the system detects active-active VSMs, one VSM is automatically reloaded based on various parameters such as VEMs attached, vCenter connectivity, last configuration time, and last active time. 2. To see any configuration changes that are performed on the rebooted VSM during the active-active condition, enter the **show system internal active-active remote accounting logs** CLI command on the active VSM.
VSM Role Collision	If another VSM is configured/provisioned with the same role (primary or secondary) in the system, the new VSM collides with the existing VSM. The **show system redundancy info** command displays the MAC addresses of the VSM(s) that collide with the working VSM.	If the problems exist, do the following: 1. Enter the **show system redundancy status** command on the VSM console. 2. Identify the VSM(s) that owns the MAC addresses that are displayed in the output of the **show system redundancy status** command. 3. Move the identified VSM(s) out of the system to stop role collision.

Symptom	Possible Causes	Solution
Both VSMs are in active mode.	Network connectivity problems. • Check for control and management VLAN connectivity between the VSM at the upstream and virtual switches. • When the VSM cannot communicate through any of these two interfaces, they will both try to become active.	If network problems exist, do the following: 1. From vSphere Client, shut down the VSM, which should be in standby mode. 2. From vSphere Client, bring up the standby VSM after network connectivity is restored.
	Different domain IDs in the two VSMs Check the *domain* value by using **show system internal redundancy info** command.	If needed, update the domain ID and save it to the startup configuration. • Upgrading the domain ID in a dual VSM system must be done as follows: – Isolate the VSM with the incorrect domain ID so that it cannot communicate with the other VSM. – Change the domain ID in the isolated VSM, save the configuration, and power off the VSM. – Reconnect the isolated VSM and power it on.

High Availability Troubleshooting Commands

This section lists commands that can be used troubleshoot problems related to high availability.

Command	Description
attach module	See Example 6-9attach module Command, page 6-10
reload module	See Example 6-8reload module Command, page 6-10
show cores	Use to list process logs and cores. See Example 6-1show cores Command, page 6-6
show processes [pid pid]	See Example 6-2show processes log [pid pid] Command, page 6-6
show system internal active-active	See Example 6-7show system internal active-active remote accounting logs Command, page 6-10

Command	Description
show system internal redundancy info	See Example 6-4show system internal redundancy info Command, page 6-7
show system internal sysmgr state	See Example 6-5show system internal sysmgr state Command, page 6-8
show system redundancy status	See Example 6-3show system redundancy status Command, page 6-6
show system redundancy status	See Example 6-6show system redundancy status Command, page 6-9

To list process logs and cores, use the following commands:

Example 6-1 show cores Command

```
switch# show cores
VDC No Module-num       Process-name      PID     Core-create-time
------ ----------       ------------      ---     ----------------
1      1                private-vlan      3207    Apr 28 13:29
```

Example 6-2 show processes log [pid pid] Command

```
switch# show processes log
VDC Process          PID     Normal-exit  Stack  Core   Log-create-time
--- ---------------  ------  -----------  -----  -----  ---------------
  1 private-vlan     3207              N      Y      N  Tue Apr 28 13:29:48 2009

switch# show processes log pid 3207
======================================================
Service: private-vlan
Description: Private VLAN

Started at Wed Apr 22 18:41:25 2009 (235489 us)
Stopped at Tue Apr 28 13:29:48 2009 (309243 us)
Uptime: 5 days 18 hours 48 minutes 23 seconds

Start type: SRV_OPTION_RESTART_STATELESS (23)
Death reason: SYSMGR_DEATH_REASON_FAILURE_SIGNAL (2) <-- Reason for the process abort
Last heartbeat 46.88 secs ago
System image name: nexus-1000v-mzg.4.0.4.SV1.1.bin
System image version: 4.0(4)SV1(1) S25

PID: 3207
Exit code: signal 6 (core dumped) <-- Indicates that a cores for the process was
generated.

CWD: /var/sysmgr/work
...
```

To check redundancy status, use the following commands:

Example 6-3 show system redundancy status Command

```
switch# show system redundancy status
Redundancy role
---------------
      administrative:   primary <-- Configured redundancy role
```

```
         operational:   primary <-- Current operational redundancy role

Redundancy mode
---------------
      administrative:   HA
         operational:   HA

This supervisor (sup-1)
-----------------------
    Redundancy state:   Active <-- Redundancy state of this VSM
    Supervisor state:   Active
      Internal state:   Active with HA standby

Other supervisor (sup-2)
------------------------
    Redundancy state:   Standby <-- Redundancy state of the other VSM
    Supervisor state:   HA standby
      Internal state:   HA standby <-- The standby VSM is in HA mode and in sync
```

To check the system internal redundancy status, use the following command:

Example 6-4 show system internal redundancy info Command

```
switch# show system internal redundancy info
My CP:
  slot: 0
  domain: 184 <-- Domain id used by this VSM
  role:   primary <-- Redundancy role of this VSM
  status: RDN_ST_AC <-- Indicates redundancy state (RDN_ST) of the this VSM is Active (AC)
  state:  RDN_DRV_ST_AC_SB
  intr:   enabled
  power_off_reqs: 0
  reset_reqs:     0
Other CP:
  slot: 1
  status: RDN_ST_SB <-- Indicates redundancy state (RDN_ST) of the other VSM is Standby
(SB)
  active: true
  ver_rcvd: true
  degraded_mode: false <-- When true, it indicates that communication through the control
interface is faulty
Redun Device 0: <-- This device maps to the control interface
  name: ha0
  pdev: ad7b6c60
  alarm: false
  mac: 00:50:56:b7:4b:59
  tx_set_ver_req_pkts:   11590
  tx_set_ver_rsp_pkts:   4
  tx_heartbeat_req_pkts: 442571
  tx_heartbeat_rsp_pkts: 6
  rx_set_ver_req_pkts:   4
  rx_set_ver_rsp_pkts:   1
  rx_heartbeat_req_pkts: 6
  rx_heartbeat_rsp_pkts: 442546 <-- Counter should be increasing, as this indicates that
communication between VSM is working properly.
  rx_drops_wrong_domain: 0
  rx_drops_wrong_slot:   0
  rx_drops_short_pkt:    0
  rx_drops_queue_full:   0
  rx_drops_inactive_cp:  0
  rx_drops_bad_src:      0
  rx_drops_not_ready:    0
  rx_unknown_pkts:       0
```

```
Redun Device 1: <-- This device maps to the mgmt interface
  name: ha1
  pdev: ad7b6860
  alarm: true
  mac: ff:ff:ff:ff:ff:ff
  tx_set_ver_req_pkts:   11589
  tx_set_ver_rsp_pkts:   0
  tx_heartbeat_req_pkts: 12
  tx_heartbeat_rsp_pkts: 0
  rx_set_ver_req_pkts:   0
  rx_set_ver_rsp_pkts:   0
  rx_heartbeat_req_pkts: 0
  rx_heartbeat_rsp_pkts: 0 <-- When communication between VSM through the control
interface is interrupted but continues through the mgmt interface, the
rx_heartbeat_rsp_pkts will increase.
  rx_drops_wrong_domain: 0
  rx_drops_wrong_slot:   0
  rx_drops_short_pkt:    0
  rx_drops_queue_full:   0
  rx_drops_inactive_cp:  0
  rx_drops_bad_src:      0
  rx_drops_not_ready:    0
  rx_unknown_pkts:       0
```

To check the system internal sysmgr state, use the following command:

Example 6-5 show system internal sysmgr state Command

```
switch# show system internal sysmgr state

The master System Manager has PID 1988 and UUID 0x1.
Last time System Manager was gracefully shutdown.
The state is SRV_STATE_MASTER_ACTIVE_HOTSTDBY entered at time Tue Apr 28 13:09:13 2009.

The '-b' option (disable heartbeat) is currently disabled.

The '-n' (don't use rlimit) option is currently disabled.

Hap-reset is currently enabled.

Watchdog checking is currently disabled.

Watchdog kgdb setting is currently enabled.


        Debugging info:

The trace mask is 0x00000000, the syslog priority enabled is 3.
The '-d' option is currently disabled.
The statistics generation is currently enabled.


        HA info:


slotid = 1    supid = 0
cardstate = SYSMGR_CARDSTATE_ACTIVE .
cardstate = SYSMGR_CARDSTATE_ACTIVE (hot switchover is configured enabled).
Configured to use the real platform manager.
Configured to use the real redundancy driver.
Redundancy register: this_sup = RDN_ST_AC, other_sup = RDN_ST_SB.
EOBC device name: eth0.
Remote addresses:  MTS - 0x00000201/3      IP - 127.1.1.2
```

```
MSYNC done.
Remote MSYNC not done.
Module online notification received.
Local super-state is: SYSMGR_SUPERSTATE_STABLE
Standby super-state is: SYSMGR_SUPERSTATE_STABLE
Swover Reason : SYSMGR_SUP_REMOVED_SWOVER <-- Reason for the last switchover
Total number of Switchovers: 0 <-- Number of switchovers
                              >> Duration of the switchover would be listed, if any.

        Statistics:

Message count:           0
Total latency:           0              Max latency:             0
Total exec:              0              Max exec:                0
```

When a role collision is detected, a warning is highlighted in the CLI output. Use the following command to display the CLI output:

Example 6-6 show system redundancy status Command

```
switch# show system redundancy status
Redundancy role
---------------
administrative: secondary
operational: secondary
Redundancy mode
---------------
administrative: HA
operational: HA
This supervisor (sup-2)
-----------------------
Redundancy state: Active
Supervisor state: Active
Internal state: Active with HA standby
Other supervisor (sup-1)
------------------------
Redundancy state: Standby
Supervisor state: HA standby
Internal state: HA standby
WARNING! Conflicting sup-2(s) detected in same domain
-----------------------------------------------------
MAC Latest Collision Time
00:50:56:97:02:3b 2012-Sep-11 18:59:17
00:50:56:97:02:3c 2012-Sep-11 18:59:17
00:50:56:97:02:2f 2012-Sep-11 18:57:42
00:50:56:97:02:35 2012-Sep-11 18:57:46
00:50:56:97:02:29 2012-Sep-11 18:57:36
00:50:56:97:02:30 2012-Sep-11 18:57:42
00:50:56:97:02:36 2012-Sep-11 18:57:46
00:50:56:97:02:2a 2012-Sep-11 18:57:36

NOTE: Please run the same command on sup-1 to check for conflicting(if any) sup-1(s) in
the same domain.
```

If no collisions are detected, the highlighted output is not displayed.

Use the following command to display the accounting logs that are stored on a remote VSM.

Example 6-7 show system internal active-active remote accounting logs Command

```
switch# show system internal active-active remote accounting logs
```

To reload a module, use the following command:

Example 6-8 reload module Command

```
switch# reload module 2
```

This command reloads the secondary VSM.

Note Entering the **reload** command without specifying a module will reload the whole system.

To attach to the standby VSM console, use the following command.

Example 6-9 attach module Command

The standby VSM console is not accessible externally, but can be accessed from the active VSM through the **attach module** *module-number* command.

```
switch# attach module 2
```

This command attaches to the console of the secondary VSM.

CHAPTER 7

VSM and VEM Modules

This chapter describes how to identify and resolve problems that relate to modules and includes the following sections:

- Information About Modules, page 7-1
- Troubleshooting a Module Not Coming Up on the VSM, page 7-1
- Problems with the VSM, page 7-4
- VSM and VEM Troubleshooting Commands, page 7-17

Information About Modules

The Cisco Nexus 1000V manages a data center defined by a VirtualCenter. Each server in the data center is represented as a module in the Cisco Nexus 1000V and can be managed as if it were a module in a physical Cisco switch.

The Cisco Nexus 1000V implementation has two parts:

- Virtual Supervisor Module (VSM)—Control software of the Cisco Nexus 1000V distributed virtual switch. It runs on a virtual machine (VM) and is based on NX-OS software.
- Virtual Ethernet Module (VEM)—Part of the Cisco Nexus 1000V that actually switches data traffic. It runs on a VMware ESX host. Several VEMs are controlled by one VSM. All the VEMs that form a switch domain should be in the same virtual data center as defined by VMware VirtualCenter.

Troubleshooting a Module Not Coming Up on the VSM

This section includes the following topics:

- Guidelines for Troubleshooting Modules, page 7-2
- Flowchart for Troubleshooting Modules, page 7-3
- Verifying the VSM Is Connected to vCenter Server, page 7-6
- Verifying the VSM Is Configured Correctly, page 7-7
- Checking the vCenter Server Configuration, page 7-10
- Checking Network Connectivity Between the VSM and the VEM, page 7-10
- Recovering Management and Control Connectivity of a Host When a VSM is Running on a VEM, page 7-12

- Checking the VEM Configuration, page 7-14
- Collecting Logs, page 7-16
- Troubleshooting L3Sec, page 8-1

Guidelines for Troubleshooting Modules

Follow these guidelines when troubleshooting a module controlled by the VSM.

- You must have a VSM VM and a VEM up and running.
- Make sure that you are running compatible versions of vCenter Server and VSM.

  For more information, see the *Cisco Nexus 1000V Compatibility Information*.
- To verify network connectivity between the VSM and vCenter Server, ping the IP address of vCenter Server. If you are using a domain name service (DNS) name, use the DNS name in the ping. If a ping to vCenter Server fails, check to see if you can ping the gateway. Otherwise, check the mgmt0 interface configuration settings.
- In the Cisco Nexus1000V Distributed Virtual Switch (DVS), we support only one vmknic with "capability l3control". If a second vmknic is added with the same capability then the host connected as VEM modules on VSM in L3 mode will go offline. To recover from this scenario, we need to remove both the vmknics from Cisco Nexus1000V DVS or migrate them back to vSwitch/VMware DVS. Once you migrate or removed, you can recreate one vmknic on Cisco Nexus1000V DVS or migrate one of the vmknic from vswitch/VMware DVS back to Cisco Nexus1000V DVS.
- Make sure that the firewall settings are OFF on the vCenter Server. If you want the firewall settings, and check to see if these ports are open:
  - Port 80
  - Port 443
- If you see the following error, verify that the VSM extension was created from vCenter Server:
  - ERROR: [VMware vCenter Server 4.0.0 build-150489]
    Extension key was not registered before its use

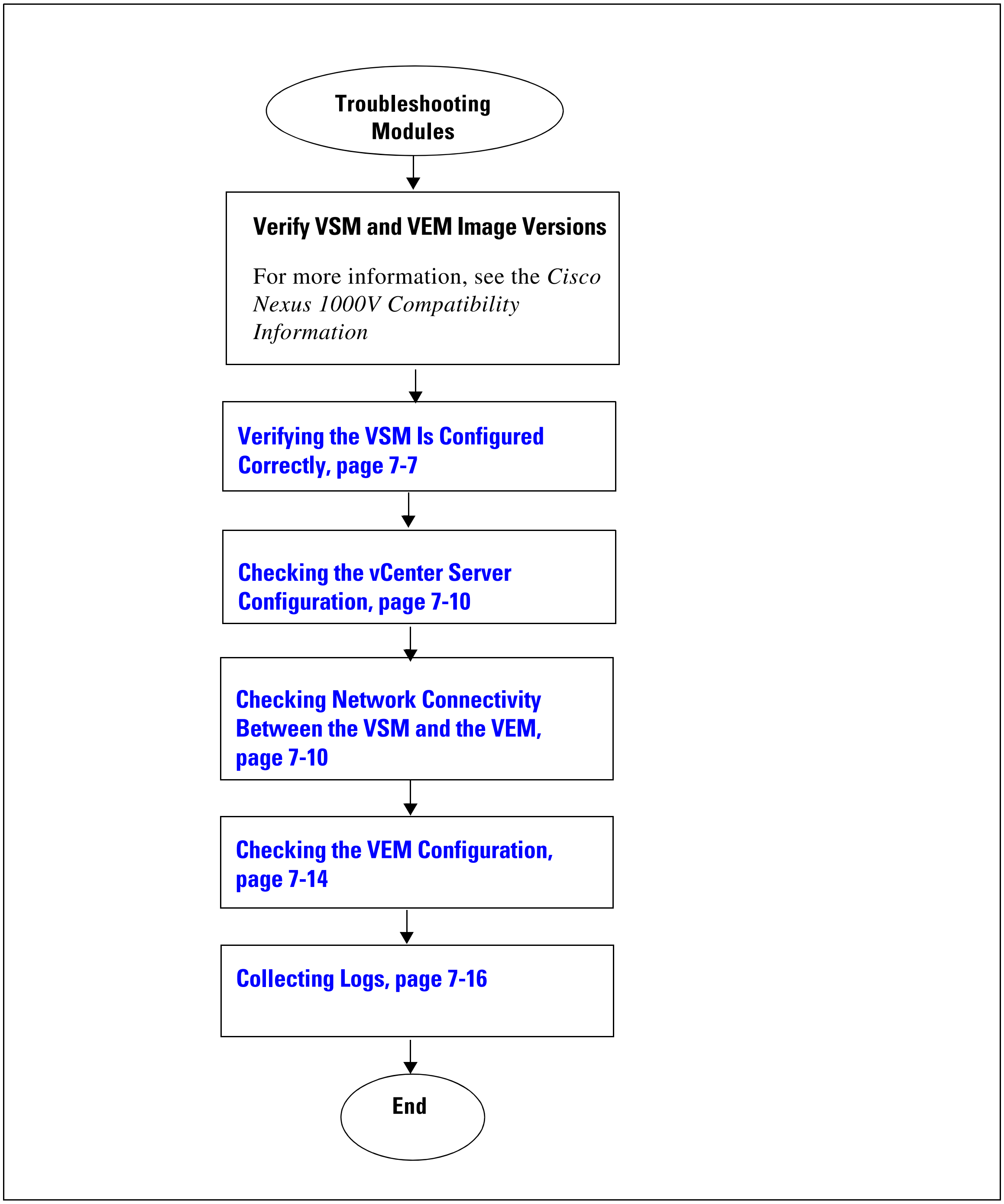
  To verity that the extension or plugin was created, see the "Finding the Extension Key on the Cisco Nexus 1000V" section on page 3-7.

  For more information about extension keys or plugins, see the "Managing Extension Keys" section on page 3-6.
- If you see the following error, see the "Checking the vCenter Server Configuration" section on page 7-10.
  - ERROR: Datacenter not found
- For a list of terms used with the Cisco Nexus 1000V, see the *Cisco Nexus 1000V Getting Started Guide*.

Flowchart for Troubleshooting Modules

Use the following flowchart to troubleshoot modules.

Flowchart: Troubleshooting Modules


Troubleshooting Modules
Verify VSM and VEM Image Versions
For more information, see the *Cisco Nexus 1000V Compatibility Information*
Verifying the VSM Is Configured Correctly, page 7-7
Checking the vCenter Server Configuration, page 7-10
Checking Network Connectivity Between the VSM and the VEM, page 7-10
Checking the VEM Configuration, page 7-14
Collecting Logs, page 7-16
End

Problems with the VSM

The following are symptoms, possible causes, and solutions for problems with the VSM.

Symptom	Possible Causes	Solution
You see the following error on the VSM: `ERROR: [VMware vCenter Server 4.0.0 build-150489] Extension key was not registered before its use`	A extension or plug-in was not created for the VSM.	**1.** Verify that the extension or plugin was created. "Finding the Extension Key Tied to a Specific DVS" procedure on page 3-8 **2.** If the plug-in is not found, create one using the following procedure in the *Cisco Nexus 1000V Getting Started Guide*: Creating a Cisco Nexus 1000V Plug-In on the vCenter Server
Following a reboot of the VSM, the system stops functioning in one of the following states and does not recover on its own. Attempts to debug fail.		
After boot, VSM is in loader prompt.	Corrupt VSM kickstart image.	**1.** Boot the VSM from the CD ROM. **2.** From the CD Boot menu, choose **Option 1, Install Nexus1000v** and bring up new image. Follow the VSM installation procedure.
	Boot variables are not set.	**1.** Boot the VSM from the CD ROM. **2.** From the CD Boot menu, choose **Option 3, Install Nexus1000v** only if the disk unformatted and bring up new image. **3.** Set the boot variables used to boot the VSM: **boot system bootflash:***system-boot-variable-name* **boot kickstart bootflash:***kickstart-boot-variable-name* **4.** Reload the VSM. **reload**
After boot, VSM is in boot prompt.	Corrupt VSM system image.	**1.** Boot the VSM from the CD ROM. **2.** From the CD Boot menu, choose **Option 1, Install Nexus1000v** and bring up new image. **3.** Follow the VSM installation procedure.

Symptom	Possible Causes	Solution
After boot, VSM is reconfigured.	Startup configuration is deleted.	Do one of the following: • If you have a saved backup copy of your configuration file, restore the configuration on the VSM. **copy** *source filesystem: filename* **system:running-config** • If not, reconfigure the VSM using the following section in the *Cisco Nexus 1000V Getting Started Guide*: Setting Up the Software
After boot, VSM is stopped at "Loader Loading."	Corrupt boot menu file.	1. Boot the VSM from the CD ROM. 2. From the CD Boot menu, choose **Option 3, Install Nexus1000v** only if the disk unformatted and bring up new image. 3. Do one of the following: • If you have a saved backup copy of your configuration file, restore the configuration on the VSM. **copy** *source filesystem: filename* **system:running-config** • If not, reconfigure the VSM using the following section in the *Cisco Nexus 1000V Getting Started Guide*: Setting Up the Software
After boot, the secondary VSM reboots continuously.	Control VLAN or control interface down	Check control connectivity between the active and the standby VSM.
	Active and standby VSMs fail to synchronize.	From the active VSM, check system manager errors to identify which application caused the failure. **show system internal sysmgr event-history errors** **show logging**
After a host reboot, the absence of a VLAN, or the wrong system VLAN on the VSM management port profile, the control and management connectivity of the VSM is lost.	The VSM is running on a VEM that it manages, but the VSM ports are not configured with system port profiles.	Run the VEM connect script locally in the ESX host where the VEM is running. Go to the VSM and configure the system VLAN in the port profile used for management. "Recovering Management and Control Connectivity of a Host When a VSM is Running on a VEM" section on page 7-12

Verifying the VSM Is Connected to vCenter Server

You can use the following procedure to verify that the VSM is connected to vCenter Server.

Step 1 Verify the connection between the VSM and vCenter Server.

show svs connections

The output should indicate that the operational status is **Connected**.

```
Example:
switch# show svs connections
connection vc:
    ip address: 172.23.231.223
    protocol: vmware-vim https
    certificate: user-installed
    datacenter name: hamilton-dc
    DVS uuid: 92 7a 14 50 05 11 15 9c-1a b0 f2 d4 8a d7 6e 6c
    config status: Disabled
    operational status: Disconnected
```

Step 2 Do one of the following:

- If the status is **Connected**, return to the "Flowchart: Troubleshooting Modules" section on page 7-3.
- If not, continue with the next step.

Step 3 Connect to vCenter Server.

config t

svs connection *connection_name*

connect

```
Example:
switch# conf t
switch(config)# svs connection HamiltonDC
switch(config-svs-conn)# connect

Example:
switch# conf t
switch(config)# svs connection HamiltonDC
switch(config-svs-conn)# connect
ERROR: [VMWARE-VIM] Extension key was not registered before its use.
```

Step 4 Do one of the following:

- If you see an error message about the Extension key, continue with the next step.
- If not, go to Step 6.

Step 5 Do the following and then go to Step 6.

- Unregister the extension key using the "Unregistering the Extension Key in the vCenter Server" section on page 3-11.
- Install a new extension key using the "Creating a Cisco Nexus 1000V Plug-In on the vCenter Server" procedure in the *Cisco Nexus 1000V Getting Started Guide*.

Step 6 Verify the connection between the VSM and vCenter Server.

show svs connections

The output should indicate that the operational status is **Connected**.

```
Example:
```

```
switch# show svs connections
connection vc:
    ip address: 172.23.231.223
    protocol: vmware-vim https
    certificate: user-installed
    datacenter name: hamilton-dc
    DVS uuid: 92 7a 14 50 05 11 15 9c-1a b0 f2 d4 8a d7 6e 6c
    config status: Disabled
    operational status: Disconnected
```

Step 7 Do one of the following:

- If the status is **Connected**, you have completed this procedure.
- If not, return to the "Flowchart: Troubleshooting Modules" section on page 7-3.

Verifying the VSM Is Configured Correctly

This section includes the following topics:

- Verifying the Domain Configuration, page 7-7
- Verifying the System Port Profile Configuration, page 7-8
- Verifying the Control and Packet VLAN Configuration, page 7-8

Verifying the Domain Configuration

You can verify the domain configuration.

BEFORE YOU BEGIN

- Log in to the CLI in EXEC mode.
- Verify that the output of the show **svs domain** command indicates the following:
  - The presence of a control VLAN and a packet VLAN.
  - The domain configuration was successfully pushed to VC.

Step 1 On the VSM, verify the domain configuration.

show svs domain

```
Example:
switch# show svs domain
SVS domain config:
  Domain id:    682
  Control vlan: 3002
  Packet vlan:  3003
  L2/L3 Control VLAN mode: L2
  L2/L3 Control VLAN interface: mgmt0
  Status: Config push to VC successful
```

Verifying the System Port Profile Configuration

You can verify the port profile configuration.

BEFORE YOU BEGIN

- Log in to the CLI in EXEC mode.
- Verify that the output of the **show port-profile name** command indicates the following:
  - The control and packet VLANs are assigned.
  - The port profile is enabled.
  - If you have configured a non-default system MTU setting, check that it is the correct size.

Step 1 On the VSM, verify the system port profile configuration.

show port-profile name *system-port-profile-name*

```
Example:
switch# show port-profile name SystemUplink
port-profile SystemUplink
  description:
  type: ethernet
  status: enabled
  capability l3control: no
  pinning control-vlan: -
  pinning packet-vlan: -
  system vlans: 114,115
  port-group: SystemUplink
  max ports: 32
  inherit:
  config attributes:
    switchport mode trunk
    switchport trunk allowed vlan all
    system mtu 1500
    no shutdown
  evaluated config attributes:
    switchport mode trunk
    switchport trunk allowed vlan all
    no shutdown
  assigned interfaces:
```

Verifying the Control and Packet VLAN Configuration

You can verify that the control and packet VLANs are configured on the VSM.

Note The procedure documented is for troubleshooting VSM and VEM connectivity with Layer 2 mode.

BEFORE YOU BEGIN

- Log in to the CLI in EXEC mode.
- Check that the output of the **show running-config** command shows control and packet VLAN ID numbers among the VLANs configured,

Step 1 On the VSM, verify that the control and packet VLANs are present.

```
switch# show running-config vlan 260-261
version 4.0(4)SV1(3)
vlan 260
  name cp_control
vlan 261
  name cp_packet

switch#
. . .
```

Step 2 Find the AIPC MAC address of the VSM on the VSM.

```
switch(config-svs-domain)# show svs neighbors

Active Domain ID: 27

AIPC Interface MAC: 0050-56bc-74f1 <-------------
inband/outband Interface MAC: 0050-56bc-62bd

Src MAC           Type   Domain-id    Node-id     Last learnt (Sec. ago)
------------------------------------------------------------------------

0050-56bc-6a3d     VSM        27         0201     771332.97
0002-3d40-1b02     VEM        27         0302        51.60
0002-3d40-1b03     VEM        27         0402        51.60
```

Step 3 Find the DPA MAC address of the VEM on the ESX host.

```
switch# vemcmd show card
Card UUID type  2: 24266920-d498-11e0-0000-00000000000f
Card name:
Switch name: Nexus1000v
Switch alias: DvsPortset-0
Switch uuid: ee 63 3c 50 04 b1 6d d6-58 61 ff ba 56 05 14 fd
Card domain: 27
Card slot: 3
VEM Tunnel Mode: L2 Mode
VEM Control (AIPC) MAC: 00:02:3d:10:1b:02
VEM Packet (inband/outband) MAC: 00:02:3d:20:1b:02
VEM Control Agent (DPA) MAC: 00:02:3d:40:1b:02 <-------------
VEM SPAN MAC: 00:02:3d:30:1b:02
Primary VSM MAC : 00:50:56:bc:74:f1
Primary VSM PKT MAC : 00:50:56:bc:62:bd
Primary VSM MGMT MAC : 00:50:56:bc:0b:d5
Standby VSM CTRL MAC : 00:50:56:bc:6a:3d
Management IPv4 address: 14.17.168.1
Management IPv6 address: 0000:0000:0000:0000:0000:0000:0000:0000
Primary L3 Control IPv4 address: 0.0.0.0
Secondary VSM MAC : 00:00:00:00:00:00
Secondary L3 Control IPv4 address: 0.0.0.0
Upgrade : Default
Max physical ports: 32
Max virtual ports: 216
Card control VLAN: 168
Card packet VLAN: 168
Control type multicast: No
Card Headless Mode : No
       Processors: 16
  Processor Cores: 8
Processor Sockets: 2
  Kernel Memory:   25102148
Port link-up delay: 5s
Global UUFB: DISABLED
```

```
Heartbeat Set: True
PC LB Algo: source-mac
Datapath portset event in progress : no
Licensed: Yes
```

Step 4 Check the upstream switches for these MAC addresses in the correct VLANs.

```
switch1 # show mac address-table | grep 1b02
* 168      0002.3d20.1b02    dynamic   20         F    F  Veth854
* 168      0002.3d40.1b02    dynamic   0          F    F  Veth854
* 1        0002.3d40.1b02    dynamic   1380       F    F  Veth854

switch2 # show mac address-table | grep 74f1
* 168      0050.56bc.74f1    dynamic   0          F    F  Eth1/1/3
```

Checking the vCenter Server Configuration

You can verify the configuration on vCenter Server.

Step 1 Confirm that the host is added to the data center and the Cisco Nexus 1000V DVS in that data center.

Step 2 Confirm that at least one pnic of the host is added to the DVS, and that pnic is assigned to the **system-uplink** profile.

Step 3 Confirm that the three VSM vnics are assigned to the port groups that contain the control VLAN, packet VLAN, and management network.

Checking Network Connectivity Between the VSM and the VEM

You can verify Layer 2 network connectivity between the VSM and the VEM.

Step 1 On the VSM, find its MAC address.

show svs neighbors

The VSM MAC address displays as the AIPC Interface MAC.

The user VEM Agent MAC address of the host displays as the Src MAC.

```
Example:
switch# show svs neighbors

Active Domain ID: 1030

AIPC Interface MAC: 0050-568e-58b7
inband/outband Interface MAC: 0050-568e-2a39

Src MAC          Type   Domain-id    Node-id     Last learnt (Sec. ago)
------------------------------------------------------------------------

0002-3d44-0602     VEM      1024         0302     261058.59
```

Step 2 Do one of the following:

- If the output of the **show svs neighbors** command in Step 1 does not display the VEM MAC address, there is a problem with connectivity between the server hosting the VSM and the upstream switch. Recheck the VSM configuration and vCenter Server configuration.
- Otherwise, continue with the next step.

Step 3 On the VEM, run the vem-health script using the VSM MAC address that you found in Step 1.

Note If the vem-health script is not in the PATH, you can find it under /usr/lib/ext/cisco/nexus/vem*/sbin/.

vem-health check *vsm_mac_address*

The vem-health script output shows the cause of the connectivity problem and recommends the next steps for troubleshooting the problem.

```
Example:
~ # vem-health check  00:50:56:a3:36:90
VSM Control MAC address: 00:50:56:a3:36:90
Control VLAN: 90
DPA MAC: 00:02:3d:40:5a:03

VSM heartbeats are not reaching the VEM.
Your uplink configuration is correct.
Recommended action:
Check if the VEM's upstream switch has learned the VSM's Control MAC.
```

Step 4 Do one of the following:

- If the VEM health check in Step 3 indicates a problem with connectivity to the upstream switch. continue with the next step.
- Otherwise, go to Step 7.

Step 5 On the upstream switch, display the MAC address table to verify the network configuration.

```
Example:
switch# show mac address-table interface Gi3/1 vlan 3002
Legend: * - primary entry
        age - seconds since last seen
        n/a - not available

  vlan   mac address     type    learn     age              ports
------+----------------+--------+-----+----------+--------------------------
Active Supervisor:
* 3002  0050.56be.7ca7   dynamic  Yes          0   Gi3/1

switch# show mac address-table interface Gi3/2 vlan 3002
Legend: * - primary entry
        age - seconds since last seen
        n/a - not available

  vlan   mac address     type    learn     age              ports
------+----------------+--------+-----+----------+--------------------------
Active Supervisor:
* 3002  00:02:3d:40:0b:0c   dynamic  Yes          0   Gi3/2
```

Step 6 Do one of the following:

- If the output from Step 5 does not display the MAC address of the VSM, then there is a problem with connectivity between the server hosting the VSM and the upstream switch. Recheck the VSM configuration and vCenter Server configuration.

- Otherwise, continue with the next step.

Step 7 On the VSM, enter the following commands to verify that the VSM MAC appears in the control and packet VLANs.

a. **config t**

b. **module vem** *module_number* **execute vemcmd show l2** *control_vlan_id*

c. **module vem** *module_number* **execute vemcmd show l2** *packet_vlan_id*

The VSM eth0 and eth1 MAC addresses should display in the host control and packet VLANs.

```
Example:
switch# config t
switch(config)# module vem 3 execute vemcmd show l2 3002
Bridge domain 3002 brtmax 100, brtcnt 3, timeout 120
    Dynamic MAC 00:50:56:be:7c:a7  LTL    16 pvlan     0 timeout   110
    Dynamic MAC 00:02:3d:40:0b:0c  LTL    10 pvlan     0 timeout   110

switch(config)# module vem 3 execute vemcmd show l2 3003
Bridge domain 3002 brtmax 100, brtcnt 3, timeout 120
    Dynamic MAC 00:50:56:be:7c:a7  LTL    16 pvlan     0 timeout   110
    Dynamic MAC 00:02:3d:20:0b:0c  LTL    10 pvlan     0 timeout   110
```

Step 8 Do one of the following:

- If the MAC address of the VSM does not appear in the output of Step 7, check the VEM configuration as explained in "Checking the VEM Configuration" section on page 7-14.
- Otherwise, you have completed this procedure.

Recovering Management and Control Connectivity of a Host When a VSM is Running on a VEM

When the VSM is running on a VEM that it manages, but the VSM ports are not configured with system port profiles, the control and management connectivity of the VSM can be lost after a host reboot or similar event. To recover from the loss, you can run the VEM connect script locally in the ESX host where the VEM is running, and then go to the VSM and configure the system VLANs in the port profile used for management.

Using the VEM Connect Script

The VEM connect script sets a given VLAN as a system VLAN on the VTEP that has the given IP address and also sets the VLAN on all the required uplinks.

If no uplink is carrying this VLAN, you also need to specify the uplink (vmnicN) on which this VLAN needs to be applied. The uplink can be a single port or a port-channel member. If it is the latter, then the script applies the VLANs as a system VLAN to all member uplinks of that port channel.

vem-connect -i *ip_address* **-v** *vlan* [**-p** *vmnicN*]

The -p parameter to the script is optional. If you run the script without the -p parameter, it tries to locate an uplink that carries this VLAN. If no such uplink exists, it reports this as an error. You need to specify the -p parameter and rerun the script.

You can recover management and control connectivity of a host when a VSM is running on a VEM.

SUMMARY

Step 1 Display the VEM ports:

vemcmd show port

```
Example:
~ # vemcmd show port
  LTL   VSM Port  Admin Link  State  PC-LTL  SGID  Vem Port  Type
   18     Eth9/2     UP   UP    F/B*    305     1    vmnic1
   20     Eth9/4     UP   UP    F/B*    305     3    vmnic3
   49      Veth1     UP   UP    FWD       0     3  VM-T-125.eth0
   50     Veth10     UP   UP    FWD       0     1      vmk1
  305        Po2     UP   UP    F/B*      0

* F/B: The port is blocked on some of the VLANs.
```

Note The output `*F/B The port is blocked on some of the VLANs` means that the trunk is not forwarding all VLANs. This situation might be normal depending on the port profile allowed VLAN list. Compare the output of the **vemcmd show port vlans** command against the list of allowed VLANs in the trunk port profile. If the lists match, all of the expected VLANs are forwarding and the Cisco Nexus 1000V is blocking nonallowed VLANs.

Step 2 Display details about the system VLANs.

vemcmd show port vlans system

```
Example:

~ # vemcmd show port vlans system
                          Native  VLAN   Allowed
  LTL   VSM Port  Mode    VLAN/   State  Vlans/SegID
                          SegID
    6   Internal   A          1   FWD    1
    8   Internal   A       3969   FWD    3969
    9   Internal   A       3969   FWD    3969
   10   Internal   A        210   FWD    210
   11   Internal   A       3968   FWD    3968
   12   Internal   A        211   FWD    211
   13   Internal   A          1   BLK    1
   14   Internal   A       3971   FWD    3971
   15   Internal   A       3971   FWD    3971
   16   Internal   A          1   FWD    1
   18     Eth9/2   T          1   FWD    210-211
   20     Eth9/4   T          1   FWD    210-211
   49      Veth1   A          1   FWD    1
   50     Veth10   A          1   FWD    1
  305        Po2   T          1   FWD    210-211
```

Step 3 Recover connectivity:

vem-connect -i *ip_address* **-v** *vlan* [**-p** *vmnicN*]

```
Example:
~ # vem-connect -i 172.23.232.67 -v 232 -p vmnic3
ltl 50 and veth Veth10 vmk1
Uplink port Po2 carries vlan 232
Set System Vlan 232 port Po2 305
Uplink port Eth9/2 carries vlan 232
Set System Vlan 232 port Eth9/2 18
Uplink port Eth9/4 carries vlan 232
```

```
Set System Vlan 232 port Eth9/4 20
Set System 232 for vmk
```

Step 4 Confirm management connectivity:

vemcmd show port vlans system

Example:

```
~ # vemcmd show port vlans system
                          Native  VLAN   Allowed
  LTL   VSM Port  Mode    VLAN/   State  Vlans/SegID
                          SegID
    6   Internal   A          1   FWD    1
    8   Internal   A       3969   FWD    3969
    9   Internal   A       3969   FWD    3969
   10   Internal   A        210   FWD    210
   11   Internal   A       3968   FWD    3968
   12   Internal   A        211   FWD    211
   13   Internal   A          1   BLK    1
   14   Internal   A       3971   FWD    3971
   15   Internal   A       3971   FWD    3971
   16   Internal   A          1   FWD    1
   18     Eth9/2   T          1   FWD    210-211,232
   20     Eth9/4   T          1   FWD    210-211,232
   49      Veth1   A          1   FWD    1
   50     Veth10   A        232   FWD    232
  305        Po2   T          1   FWD    210-211,232
```

Checking the VEM Configuration

You can verify that the ESX host received the VEM configuration and setup.

Step 1 On the ESX host, confirm that the VEM Agent is running and that the correct host uplinks are added to the DVS.

vem status

Example:

```
~ # vem status
VEM modules are loaded

Switch Name    Num Ports   Used Ports  Configured Ports  MTU     Uplinks
vSwitch0       64          3           64                1500    vmnic0
DVS Name       Num Ports   Used Ports  Configured Ports  Uplinks
switch         256         9           256               vmnic1 VEM Agent is running
```

Step 2 Restore connectivity that is lost due to an incorrect MTU value on an uplink.

a. **vemcmd show port** *port-LTL-number*

b. **vemcmd set mtu** *value* **ltl** *port-LTL-number*

Example:

```
~ # vemcmd show port 48
LTL    IfIndex   Vlan    Bndl  SG_ID Pinned_SGID  Type  Admin State  CBL Mode Name
. . .
17   1a030100      1 T   304      1           32  PHYS     UP    UP    1  Trunk vmnic1
~# vemcmd set mtu 9000 ltl 17
```

Note Use these **vemcmds** only as a recovery measure and then update the MTU value in the port profile configuration for system uplinks or in the interface configuration for nonsystem uplinks.

Step 3 Verify that the domain ID, control VLANs, and packet VLANs are configured correctly on the host.

vemcmd show card

```
Example:
~ # vemcmd show card
Card UUID type  2: 58f8afd7-e1e3-3c51-85e2-6e6f2819a7b8
Card name: sfish-srvr-1
Switch name: switch
Switch alias: DvsPortset-0
Switch uuid: 56 e0 36 50 91 1c 32 7a-e9 9f 31 59 88 0c 7f 76
Card domain: 1024
Card slot: 4
VEM Control (Control VLAN) MAC: 00:02:3d:14:00:03
VEM Packet (inband/outband) MAC: 00:02:3d:24:00:03
VEM Control Agent (DPA) MAC: 00:02:3d:44:00:03
VEM SPAN MAC: 00:02:3d:34:00:03
Management IP address: 172.23.232.102
Max physical ports: 32
Max virtual ports: 216
Card control VLAN: 3002
Card packet VLAN: 3003
       Processors: 4
  Processor Cores: 4
Processor Sockets: 2
  Physical Memory: 4290351104
```

Step 4 Verify that the ports of the host added to the DVS are listed and that the ports are correctly configured as access or trunk on the host.

vemcmd show port

```
Example:
~ # vemcmd show port
LTL    IfIndex   Vlan    Bndl  SG_ID Pinned_SGID  Type  Admin State  CBL Mode   Name
8          0   3969       0      2           2  VIRT     UP    UP    1 Access l20
9          0   3969       0      2           2  VIRT     UP    UP    1 Access l21
10         0   3002       0      2           2  VIRT     UP    UP    1 Access l22
11         0   3968       0      2           2  VIRT     UP    UP    1 Access l23
12         0   3003       0      2           2  VIRT     UP    UP    1 Access l24
13         0      1       0      2           2  VIRT     UP    UP    0 Access l25
14         0   3967       0      2           2  VIRT     UP    UP    1 Access l26
16  1a030100      1 T     0      2           2  PHYS     UP    UP    1 Trunk vmnic1
```

The last line of output indicates that vmnic1 should be in Trunk mode, with the CBL value of 1. The CBL value of the native VLAN does not have to be 1. It will be 0 if it is not allowed, or 1 if it is VLAN 1 and not allowed. This issue is not a problem unless the native VLAN is the Control VLAN.The Admin state and Port state should be UP.

Step 5 Verify that the vmnic port that is supposed to carry the control VLAN and packet VLAN is present.

vemcmd show bd *control_vlan*
vemcmd show bd *packet_vlan*

```
Example:
~ # vemcmd show bd 3002
BD 3002, vdc 1, vlan 3002, 2 ports
Portlist:
     10  l22
     16  vmnic1
~ # vemcmd show bd 3003
BD 3003, vdc 1, vlan 3003, 2 ports
Portlist:
     12  l24
     16  vmnic1
```

Step 6 Verify the following:

- The control and packet VLANs are shown in the command output, indicating that the DV port groups are successfully pushed from vCenter Server to the host.
- The correct physical trunk port vmnic is used.

vemcmd show trunk

```
Example:
~ # vemcmd show trunk
Trunk port 16 native_vlan 1 CBL 1vlan(1) cbl 1, vlan(3002) cbl 1, vlan(3003) cbl 1,
```

At least one physical uplink must be carrying the control and packet VLANs. If more than one uplink is carrying the control and packet VLANs, the uplinks must be in a port channel profile. The port channel itself would not be visible because the VEM is not yet added to the VSM.

Step 7 Restore connectivity that is lost due to incorrect port and system VLAN settings.

vemcmd show port *port-ltl-number*

vemcmd set system-vlan *vlan_id* **ltl** *port-ltl-number*

```
Example:
~ # vemcmd show port 48
LTL    IfIndex   Vlan    Bndl  SG_ID Pinned_SGID  Type  Admin State  CBL Mode   Name
. . .
48     1b030000     1       0     32           1 VIRT     UP  DOWN     0 Access vmk1
~ # vemcmd set system-vlan 99 ltl 48
```

Note Use these **vemcmds** only as a recovery measure and then update the port profile configuration with correct system VLANs.

Collecting Logs

After you have verified network connectivity between the VEM and the VSM, you can use the following procedure to collect log files to help identify the problem.

Step 1 On the VEM, verify its UUID.

vemcmd show card info

```
Example:
~ # vemcmd show card info
Card UUID type  0: 4908a717-7d86-d28b-7d69-001a64635d18
Card name: sfish-srvr-7
Switch name: switch
Switch uuid: 50 84 06 50 81 36 4c 22-9b 4e c5 3e 1f 67 e5 ff
Card domain: 11
Card slot: 12
Control VLAN MAC: 00:02:3d:10:0b:0c
inband/outband MAC: 00:02:3d:20:0b:0c
SPAN MAC: 00:02:3d:30:0b:0c
USER DPA MAC: 00:02:3d:40:0b:0c
Management IP address: 172.28.30.56
Max physical ports: 16
Max virtual ports: 32
Card control VLAN: 3002
Card packet VLAN: 3003
```

Step 2 On the VSM, verify the module number to which the corresponding UUID entry is mapped.

show module vem mapping

```
Example:
switch# show module vem mapping
Mod     Status          UUID                                    License Status
---     -----------     ------------------------------------    --------------
60      absent          33393935-3234-5553-4538-35314e355400    unlicensed
66      powered-up      33393935-3234-5553-4538-35314e35545a    licensed
switch#
```

Step 3 Using the module number from Step 2, collect the output of the following commands:

- **show system internal vem_mgr event-history module 13**
- **show module internal event-history module 13**
- **show system internal im event-history module 13**
- **show system internal vmm event-history module 13**
- **show system internal ethpm event-history module 13**

Note If you need to contact Cisco TAC for assistance in resolving an issue, you will need the output of the commands listed in Step 3.

VSM and VEM Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to VSM.

Command	Description
show svs neighbors	Displays all neighbors. See Example 7-1 on page 7-19.
show svs connections	Displays the Cisco Nexus 1000V connections. See Example 7-2 on page 7-19.
show svs domain	Displays the domain configuration. See Example 7-3 on page 7-19.

Command	Description
show port-profile name *name*	Displays the configuration for a named port profile. See Example 7-4 on page 7-20.
show running-config vlan *vlanID*	Displays the VLAN information in the running configuration. See Example 7-5 on page 7-20.
vem-health check *vsm_mac_address*	Displays the cause of a connectivity problem and recommends how to troubleshoot the problem. See Example 7-6 on page 7-20.
show mac address-table interface	Displays the MAC address table on an upstream switch to verify the network configuration. See Example 7-7 on page 7-20.
module vem *module_number* **execute vemcmd show l2** [*control_vlan_id* | *packet_vlan_id*]	Displays the VLAN configuration on the VEM to verify that the VSM MAC appears in the control and packet VLANs. See Example 7-8 on page 7-21.
vem status	Displays the VEM status to confirm that the VEM Agent is running and that the correct host uplinks are added to the DVS. See Example 7-9 on page 7-21.
vemcmd show card	Displays information about cards on the VEM to verify that the domain ID, control VLANs, and packet VLANs are configured correctly on the host. See Example 7-10 on page 7-21.
vemcmd show port [*port-LTL-number*]	Displays information about ports on the VEM to verify that the ports of the host added to the DVS are listed and that the ports are correctly configured as access or trunk on the host. See Example 7-11 on page 7-21. See Example 7-12 on page 7-22.
vemcmd show bd [*control_vlan_id* | *packet_vlan_id*]	Displays configured information on the VEM to verify that the VM NIC port that is supposed to carry the control VLAN and packet VLAN is present. See Example 7-15 on page 7-22.
vemcmd show trunk	Displays configured information on the VEM to verify that the DV port groups are successfully pushed from vCenter Server to the host and that the correct physical trunk port VM NIC is used. See Example 7-16 on page 7-22.
vem-connect -i *ip_address* **-v** *vlan* **[-pnic** *vmnicN*]	Recovers management and control connectivity of a host when a VSM is running on a VEM.

Command	Description
show module vem mapping	Displays information about the VEM that a VSM maps to, including the VEM module number, status, UUID, and license status. See Example 7-17 on page 7-22.
show system internal vem_mgr event-history module 13 *module-number*	Displays module FSM event information.
show module internal event-history module *module-number*	Displays the event log for a module.
show system internal im event-history module *module-number*	Displays the module IM event logs for the system.
show system internal vmm event-history module *module-number*	Displays the module VMM event logs for the system.
show system internal ethpm event-history module *module-number*	Displays the module Ethernet event logs for the system.
show system internal ethpm event-history int *type slot*	Displays the Ethernet interface logs for the system.

Example 7-1 show svs neighbors Command

```
switch# show svs neighbors

Active Domain ID: 113

AIPC Interface MAC: 0050-56b6-2bd3
inband/outband Interface MAC: 0050-56b6-4f2d

Src MAC           Type   Domain-id    Node-id     Last learnt (Sec. ago)
------------------------------------------------------------------------

0002-3d40-7102     VEM       113         0302     71441.12
0002-3d40-7103     VEM       113         0402       390.77

switch#
```

Example 7-2 show svs connections Command

```
switch# show svs connections
connection vc:
    ip address: 172.23.231.223
    protocol: vmware-vim https
    certificate: user-installed
    datacenter name: hamilton-dc
    DVS uuid: 92 7a 14 50 05 11 15 9c-1a b0 f2 d4 8a d7 6e 6c
    config status: Disabled
    operational status: Disconnected
```

Example 7-3 show svs domain Command

```
switch# show svs domain
SVS domain config:
  Domain id:    682
  Control vlan: 3002
  Packet vlan:  3003
  L2/L3 Control VLAN mode: L2
  L2/L3 Control VLAN interface: mgmt0
  Status: Config push to VC successful
```

Example 7-4 show port-profile Command

```
switch# show port-profile name SystemUplink
port-profile SystemUplink
  description:
  type: ethernet
  status: enabled
  capability l3control: no
  pinning control-vlan: -
  pinning packet-vlan: -
  system vlans: 114,115
  port-group: SystemUplink
  max ports: 32
  inherit:
  config attributes:
    switchport mode trunk
    switchport trunk allowed vlan all
    system mtu 1500
    no shutdown
  evaluated config attributes:
    switchport mode trunk
    switchport trunk allowed vlan all
    no shutdown
  assigned interfaces:
```

Example 7-5 show running-configuration vlan Command

```
switch# show running-config vlan 260-261
version 4.0(4)SV1(3)
vlan 260
  name cp_control
vlan 261
  name cp_packet

switch#
```

Example 7-6 vem-health check Command

```
~ # vem-health check  00:50:56:a3:36:90
VSM Control MAC address: 00:50:56:a3:36:90
Control VLAN: 90
DPA MAC: 00:02:3d:40:5a:03

VSM heartbeats are not reaching the VEM.
Your uplink configuration is correct.
Recommended action:
Check if the VEM's upstream switch has learned the VSM's Control MAC.
```

Example 7-7 show mac address-table interface Command

```
switch# show mac address-table interface Gi3/1 vlan 3002
Legend: * - primary entry
        age - seconds since last seen
        n/a - not available

  vlan   mac address     type    learn     age              ports
------+----------------+--------+-----+----------+--------------------------
```

```
Active Supervisor:
* 3002  0050.56be.7ca7   dynamic  Yes          0   Gi3/1
```

Example 7-8 module vem execute vemcmd show l2 Command

```
switch# config t
switch(config)# module vem 3 execute vemcmd show l2 3002
Bridge domain 3002 brtmax 100, brtcnt 3, timeout 120
    Dynamic MAC 00:50:56:be:7c:a7  LTL    16 pvlan     0 timeout   110
    Dynamic MAC 00:02:3d:40:0b:0c  LTL    10 pvlan     0 timeout   110

switch(config)# module vem 3 execute vemcmd show l2 3003
Bridge domain 3002 brtmax 100, brtcnt 3, timeout 120
    Dynamic MAC 00:50:56:be:7c:a7  LTL    16 pvlan     0 timeout   110
    Dynamic MAC 00:02:3d:20:0b:0c  LTL    10 pvlan     0 timeout   110
```

Example 7-9 vem status Command

```
~ # vem status
VEM modules are loaded

Switch Name    Num Ports   Used Ports  Configured Ports  MTU     Uplinks
vSwitch0       64          3           64                1500    vmnic0
DVS Name       Num Ports   Used Ports  Configured Ports  Uplinks
switch         256         9           256               vmnic1 VEM Agent is running
```

Example 7-10 vemcmd show card Command

```
~ # vemcmd show card
Card UUID type  2: 58f8afd7-e1e3-3c51-85e2-6e6f2819a7b8
Card name: sfish-srvr-1
Switch name: switch
Switch alias: DvsPortset-0
Switch uuid: 56 e0 36 50 91 1c 32 7a-e9 9f 31 59 88 0c 7f 76
Card domain: 1024
Card slot: 4
VEM Control (Control VLAN) MAC: 00:02:3d:14:00:03
VEM Packet (inband/outband) MAC: 00:02:3d:24:00:03
VEM Control Agent (DPA) MAC: 00:02:3d:44:00:03
VEM SPAN MAC: 00:02:3d:34:00:03
Management IP address: 172.23.232.102
Max physical ports: 32
Max virtual ports: 216
Card control VLAN: 3002
Card packet VLAN: 3003
       Processors: 4
  Processor Cores: 4
Processor Sockets: 2
  Physical Memory: 4290351104
```

Example 7-11 vemcmd show port Command

```
~ # vemcmd show port
LTL    IfIndex   Vlan    Bndl  SG_ID Pinned_SGID  Type  Admin State  CBL Mode    Name
8          0   3969       0      2           2  VIRT     UP    UP    1 Access l20
9          0   3969       0      2           2  VIRT     UP    UP    1 Access l21
10         0   3002       0      2           2  VIRT     UP    UP    1 Access l22
11         0   3968       0      2           2  VIRT     UP    UP    1 Access l23
12         0   3003       0      2           2  VIRT     UP    UP    1 Access l24
13         0      1       0      2           2  VIRT     UP    UP    0 Access l25
14         0   3967       0      2           2  VIRT     UP    UP    1 Access l26
```

```
16  1a030100      1 T     0      2           2  PHYS     UP    UP    1 Trunk vmnic1
```

Example 7-12 vemcmd show port Command

```
~ # vemcmd show port 48
LTL    IfIndex   Vlan    Bndl  SG_ID Pinned_SGID  Type  Admin State  CBL Mode
Name  . . .
17   1a030100      1 T   304      1          32  PHYS     UP    UP    1  Trunk vmnic1
```

Example 7-13 vemcmd show port Command

```
~ vemcmd show port
  LTL   VSM Port  Admin Link  State  PC-LTL  SGID  Vem Port
   17     Eth5/1     UP   UP    FWD     305     0  vmnic0
   18     Eth5/2     UP   UP    FWD     305     1  vmnic1
   49     Veth11     UP   UP    FWD       0     0  vmk0
   50     Veth14     UP   UP    FWD       0     1  vmk1
   51     Veth15     UP   UP    FWD       0     0  vswif0
  305        Po1     UP   UP    FWD       0

* F/B: Port is BLOCKED on some of the vlans.
 Please run "vemcmd show port vlans" to see the details.
```

Example 7-14 vemcmd show port vlans Command

```
~ # vemcmd show port vlans
                        Native  VLAN   Allowed
  LTL   VSM Port  Mode  VLAN    State  Vlans
   17     Eth5/1   T        1   FWD    1,100,119,219,319
   18     Eth5/2   T        1   FWD    1,100,119,219,319
   49     Veth11   A      119   FWD    119
   50     Veth14   A      119   FWD    119
   51     Veth15   A      119   FWD    119
  305        Po1   T        1   FWD    1,100,119,219,319
```

Note The output `*F/B The port is blocked on some of the VLANs` means that the trunk is not forwarding all VLANs. This might be a normal situation depending on the port profile allowed VLAN list. Compare the output of the **vemcmd show port vlans** command against the port profile trunk allowed VLANs. If the lists match, all of the expected VLANs are forwarding and the Cisco Nexus 1000V is blocking nonallowed VLANs.

Example 7-15 vemcmd show bd Command

```
~ # vemcmd show bd 3002
BD 3002, vdc 1, vlan 3002, 2 ports
Portlist:
     10  l22
     16  vmnic1
```

Example 7-16 vemcmd show trunk Command

```
~ # vemcmd show trunk
Trunk port 16 native_vlan 1 CBL 1vlan(1) cbl 1, vlan(3002) cbl 1, vlan(3003) cbl 1,
```

Example 7-17 show module vem mapping Command

```
switch# show module vem mapping
Mod     Status          UUID                                    License Status
```

```
---     -----------     ------------------------------------    --------------
60      absent          33393935-3234-5553-4538-35314e355400    unlicensed
66      powered-up      33393935-3234-5553-4538-35314e35545a    licensed
switch#
```

CHAPTER 8

L3Sec

This chapter describes how to secure the internal control plane communications (Control and Packet traffic) of Nexus 1000V in a more robust way than in previous releases. It operates only in Layer 3 Control mode.

- Troubleshooting L3Sec, page 8-1

Troubleshooting L3Sec

The following are symptoms, possible causes and solutions identified while troubleshooting L3Sec.

Symptom

Table 8-1 Troubleshooting L3Sec

Possible Causes	Solution
SVS connection is not up.	1. Verify SVS connection. **Show svs connection** 2. If the connection is "not connected", do connect
Key mismatch between VSM / VEM.	1. Verify key fields mismatch between switch opaque data and vem. 2. Do, show vms internal info dvs and check the keys present. 3. On vem, perform "vemcmd show sod" and check if the fields chunk1, chunk2 and chunk3 are matching. 4. If mismatches, disable and enable l3sec again using "[no] enable l3sec" under svs-domain.
Boot variables are not set.	1. Verify running config. **Show running config** 2. If "enable l3sec" is present under svs-domain. 3. If not present, do "enable l3sec" and check for any error messages and perform action accordingly.

CHAPTER 9

Ports

This chapter describes how to identify and resolve problems with ports and includes the following sections:

Information About Ports

This section includes the following topics:

Information About Interface Characteristics

Before a switch can relay frames from one data link to another, you must define the characteristics of the interfaces through which the frames are received and sent. The configured interfaces can be Ethernet (physical) interfaces, virtual Ethernet interfaces, and the management interface.

Each interface has the following:

- Administrative Configuration

  The administrative configuration does not change unless you modify it. This configuration has attributes that you can configure in administrative mode.

- Operational state

  The operational state of a specified attribute, such as the interface speed. This state cannot be changed and is read-only. Some values might not be valid when the interface is down (such as the operation speed).

For a complete description of port modes, administrative states, and operational states, see the *Cisco Nexus 1000V Interface Configuration Guide*.

Information About Interface Counters

Port counters are used to identify synchronization problems. Counters can show a significant disparity between received and transmitted frames. To display interface counters, use the following command:

show interface ethernet *slot number* **counters**

See Example 9-11 on page 9-14.

Values stored in counters can be meaningless for a port that has been active for an extended period. Clearing the counters provides a better idea of the actual link behavior at the present time. Create a baseline first by clearing the counters.

clear counters interface ethernet *slot-number*

Information About Link Flapping

When a port continually goes up and down, it is said to be flapping, or link flapping. When a port is flapping, it cycles through the following states, in this order, and then starts over again:

1. Initializing—The link is initializing.
2. Offline—The port is offline.
3. Link failure or not connected—The physical layer is not operational and there is no active device connection.

To troubleshoot link flapping, see the "Information About Link Flapping" section on page 9-2.

Information About Port Security

The port security feature allows you to secure a port by limiting and identifying the MAC addresses that can access the port. Secure MAC addresses can be manually configured or dynamically learned.

For detailed information about port security, see the *Cisco Nexus 1000V Security Configuration Guide*.

Type of Port	Is Port Security Supported?
vEthernet access	Yes
vEthernet trunk	Yes
vEthernet SPAN destination	No
Standalone Ethernet interfaces	No
Port channel members	No

To troubleshoot problems with port security, see the following:

- VM Cannot Ping a Secured Port, page 9-7
- Port Security Violations, page 9-8

Port Diagnostic Checklist

Use the following checklist to diagnose port interface activity.

For more information about port states, see the *Cisco Nexus 1000V Interface Configuration Guide*.

Table 9-1 Port Diagnostic Checklist

Checklist	Example	√
Verify that the module is active. **show module**	See Example 9-1 on page 9-11.	
Verify that the VSM is connected to vCenter Server. **show svs connections**	See Example 9-3 on page 9-12.	
On vSphere Client connected to vCenter Server, verify that the required port profiles are assigned to the physical NICs and the virtual NICs.		
Verify that the ports have been created. **show interface brief**	See Example 9-8 on page 9-13.	
Verify the state of the interface. **show interface ethernet**	See Example 9-10 on page 9-13.	

Problems with Ports

This section includes possible causes and solutions for the following symptoms:

- Cannot Enable an Interface, page 9-4
- Port Link Failure or Port Not Connected, page 9-4
- Link Flapping, page 9-5
- Port ErrDisabled, page 9-6
- VM Cannot Ping a Secured Port, page 9-7
- Port Security Violations, page 9-8
- Port State is Blocked on a VEM, page 9-9

Cannot Enable an Interface

Possible Cause	Solution
A Layer 2 port is not associated with an access VLAN or the VLAN is suspended.	1. Verify that the interface is configured in a VLAN. **show interface brief** 2. If not already, associate the interface with an access VLAN. 3. Determine the VLAN status. **show vlan brief** 4. If not already active, configure the VLAN as active. **config t** **vlan** *vlan-id* **state active**

Port Link Failure or Port Not Connected

Possible Cause	Solution
The port connection is bad.	1. Verify the port state. **show system internal ethpm info** 2. Disable and then enable the port. **shut** **no shut** 3. Move the connection to a different port on the same module or a different module. 4. Collect the ESX-side NIC configuration. **vss-support**
The link is stuck in initialization state or the link is in a point-to-point state.	1. Check for a link failure system message. `Link Failure, Not Connected` **show logging** 2. Disable and then enable the port. **shut** **no shut** 3. Move the connection to a different port on the same module or a different module. 4. Collect the ESX-side NIC configuration. **vss-support**

Link Flapping

When you are troubleshooting unexpected link flapping, it is important to have the following information:

- Who initiated the link flap.
- The actual reason for the link being down.
- For a definition of link flapping, see the "Link Flapping" section on page 9-5.

Possible Cause	Solution
The bit rate exceeds the threshold and puts the port into an error-disabled state.	Disable and then enable the port. **shut** **no shut** The port should return to the normal state.
A hardware failure or intermittent hardware error causes a packet drop in the switch.	An external device might choose to initialize the link again when encountering the error. If so, the exact method of link initialization varies by device. **1.** Determine the reason for the link flap as indicated by the MAC driver. **2.** Use the debug facilities on the end device to troubleshoot the problem.
A software error causes a packet drop.	
A control frame is erroneously sent to the device.	
ESX errors, or link flapping, occurs on the upstream switch.	Use the troubleshooting guidelines in the documentation for your ESX or upstream switch.

Port ErrDisabled

Possible Cause	Solution
The cable is defective or damaged.	1. Verify the physical cabling. 2. Replace or repair defective cables. 3. Reenable the port. **shut** **no shut**
You attempted to add a port to a port channel that was not configured identically, and the port is then errdisabled.	1. Display the switch log file and identify the exact configuration error in the list of port state changes. **show logging logfile** 2. Correct the error in the configuration and add the port to the port channel. 3. Re-enable the port. **shut** **no shut**
A VSM application error has occurred.	1. Identify the component that had an error while you were bringing up the port. **show logging logfile | grep** *interface_number* See Example 9-7 on page 9-13. 2. Identify the error transition. **show system internal ethpm event-history interface** *interface_number* 3. Open a support case and submit the output of the above commands. For more information see the “Contacting Cisco or VMware Customer Support” section on page 1-7.

VM Cannot Ping a Secured Port

Possible Cause	Solution
The vEthernet interface is not up.	1. Verify the state of the vEthernet interface. **show interface vethernet** *number* 2. If the interface is down, enable it. **shut** **no shut**
Drop on Source Miss (DSM) is set. New MAC addresses cannot be learned by this port.	1. Verify the port security configuration. **module vem 3 execute vemcmd show portsec stats** 2. If DSM is set, clear the DSM bit on the VSM. **no port-security stop learning**
The packet VLAN is not allowed on the port.	1. Identify the packet VLAN ID. **show svs domain** 2. Verify that the packet VLAN is allowed on VEM uplink ports. **show port-profile na uplink-all** 3. If the packet VLAN is not allowed on the uplink port profile, add it to the allowed VLAN list.
The packet VLAN is not allowed on the upstream switch port.	1. Identify the upstream neighbors connected to the interface. **show cdp neighbors** 2. Log in to the upstream switch and verify that the packet VLAN is allowed on the port. **show running-config interface gigabitEthernet** *slot/port* 3. If the packet VLAN is not allowed on the port, add it to the allowed VLAN list.

Port Security Violations

For detailed information about port security, see the *Cisco Nexus 1000V Security Configuration Guide*.

<table>
<tr><th>Possible Cause</th><th>Solution</th></tr>
<tr><td>The configured maximum number of secured addresses on the port is exceeded.</td><td>1. Display the secure addresses.
show port -security address vethernet *number*
show port-security address interface vethernet number
2. Identify ports with a security violation.
show logging | inc "PORT-SECURITY-2-ETH_PORT_SEC_SECURITY_VIOLATION_MAX_MAC_VLAN"
3. Correct the security violation.
4. Enable the interface.
shut
no shut</td></tr>
</table>

Port State is Blocked on a VEM

Possible Cause	Solution
The VLAN is not created on the VSM.	1. Verify the status and of the vEthernet interface. It should be up and not inactive. **show interface vethernet number** 2. Verify that the VLAN on the VSM is created. **show vlan** *vlan-id* On the VEM module, do the following: 1. Verify that the VLAN is programmed. **vemcmd show vlan** *vlan-id* 2. Verify that the VLAN is allowed on the ports. **vemcmd show port vlan** 3. Create the VLAN on the VSM. **vlan** *vlan-id*
The VEM modules are unlicensed.	1. Verify that all the modules are in licensed state. **show module** 2. Verify the status of the vEthernet interface. It should be up and not "VEM Unlicensed.” **show interface vethernet number** 3. Verify the license status of VEM modules. **show module vem license-info** On the VEM module, do the following: 1. Verify that card details show Licensed: Yes. **vemcmd show card** 2. Install the necessary licenses or move the switch to essential mode. **svs switch edition essential**

Port Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to ports.

Command	Purpose
show module *module-number*	Displays the state of a module. See Example 9-1 on page 9-11.
show svs domain	Displays the domain configuration. See Example 9-2 on page 9-12.
show svs connections	Displays the Cisco Nexus 1000V connections. See Example 9-3 on page 9-12.
show cdp neighbors	Displays the neighbors connected to an interface. See Example 9-4 on page 9-12.
show port internal event-history interface	Displays information about the internal state transitions of the port. See Example 9-5 on page 9-12.
show logging logfile	Displays logged system messages. See Example 9-6 on page 9-12.
show logging logfile | grep *interface_number*	Displays logged system messages for a specified interface. See Example 9-7 on page 9-13.
show interface brief	Displays a table of interface states. See Example 9-8 on page 9-13.
show interface ethernet	Displays the configuration for a named Ethernet interface, including the following: • Administrative state • Speed • Trunk VLAN status • Number of frames sent and received • Transmission errors, including discards, errors, CRCs, and invalid frames See Example 9-9 on page 9-13. See Example 9-10 on page 9-13.

Command	Purpose
show interface ethernet counters	Displays port counters for identifying synchronization problems. For information about counters, see the "Information About Interface Counters" section on page 9-2. See Example 9-11 on page 9-14.
show interface vethernet	Displays the vEthernet interface configuration. See Example 9-12 on page 9-14.
show interface status	Displays the status of the named interface.
show interface capabilities	Displays a tabular view of all configured port profiles. See Example 9-13 on page 9-14.
show interface virtual port mapping	Displays the virtual port mapping for all vEthernet interfaces. See Example 9-14 on page 9-16.
module vem execute vemcmd show portsec status	Displays the port security status of the port. If enabled, the output shows an LTL connected to the VM network adapter. See Example 9-15 on page 9-16.
show port-security interface veth	Displays secure vEthernet interfaces.
show port -security address interface vethernet	Displays information about secure addresses on an interface. See Example 9-17 on page 9-16.

For detailed information about **show** command output, see the *Cisco Nexus 1000V Command Reference*.

EXAMPLES

Example 9-1 show module Command

```
switch# show mod 3
Mod  Ports  Module-Type                      Model              Status
---  -----  -------------------------------- ------------------ ------------
3    248    Virtual Ethernet Module                             ok

Mod  Sw              Hw
---  --------------  ------
3    NA              0.0

Mod  MAC-Address(es)                         Serial-Num
---  --------------------------------------  ----------
```

```
3    02-00-0c-00-03-00 to 02-00-0c-00-03-80  NA

Mod  Server-IP        Server-UUID                           Server-Name
---  ---------------  ------------------------------------  --------------------
3    192.168.48.20    496e48fa-ee6c-d952-af5b-001517136344  frodo
```

Example 9-2 show svs domain Command

```
switch# show svs domain
SVS domain config:
  Domain id:    559
  Control vlan: 3002
  Packet vlan:  3003
  L2/L3 Aipc mode: L2
  L2/L3 Aipc interface: management interface0
  Status: Config push to VC successful.
switch#
```

Example 9-3 show svs connections Command

```
switch# show svs connections
connection VC:
    ip address: 192.168.0.1
    protocol: vmware-vim https
    certificate: default
    datacenter name: Hamilton-DC
    DVS uuid: ac 36 07 50 42 88 e9 ab-03 fe 4f dd d1 30 cc 5c
    config status: Enabled
    operational status: Connected
switch#
```

Example 9-4 show cdp neighbors Command

```
switch# show cdp neighbors
Capability Codes: R - Router, T - Trans-Bridge, B - Source-Route-Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  V - VoIP-Phone, D - Remotely-Managed-Device,
                  s - Supports-STP-Dispute

Device ID              Local Intrfce   Hldtme  Capability  Platform      Port ID
swordfish-6k-2         Eth3/2        149    R S I       WS-C6506-E    Gig1/38
switch#
```

Example 9-5 show port internal event-history interface Command

```
switch# show port internal event-history interface e1/7
>>>>FSM: <e1/7> has 86 logged transitions<<<<<
1) FSM:<e1/7> Transition at 647054 usecs after Tue Jan  1 22:44..
    Previous state: [PI_FSM_ST_IF_NOT_INIT]
    Triggered event: [PI_FSM_EV_MODULE_INIT_DONE]
    Next state: [PI_FSM_ST_IF_INIT_EVAL]
2) FSM:<e1/7> Transition at 647114 usecs after Tue Jan  1 22:43..
    Previous state: [PI_FSM_ST_IF_INIT_EVAL]
    Triggered event: [PI_FSM_EV_IE_ERR_DISABLED_CAP_MISMATCH]
    Next state: [PI_FSM_ST_IF_DOWN_STATE]
```

Example 9-6 show logging logfile Command

```
switch# show logging logfile
 . . .
Jan  4 06:54:04 switch %PORT_CHANNEL-5-CREATED: port-channel 7 created
Jan  4 06:54:24 switch %PORT-5-IF_DOWN_PORT_CHANNEL_MEMBERS_DOWN: Interface port-channel 7
is down (No operational members)
Jan  4 06:54:40 switch %PORT_CHANNEL-5-PORT_ADDED: e1/8 added to port-channel 7
Jan  4 06:54:56 switch %PORT-5-IF_DOWN_ADMIN_DOWN: Interface e1/7 is down (Admnistratively
down)
Jan  4 06:54:59 switch %PORT_CHANNEL-3-COMPAT_CHECK_FAILURE: speed is not compatible
Jan 4 06:55:56 switch%PORT_CHANNEL-5-PORT_ADDED: e1/7 added to port-channel 7
switch#
```

Example 9-7 show logging logfile | grep Command

```
switch# show logging logfile | grep Vethernet3626
2011 Mar 25 10:56:03 n1k-bl %VIM-5-IF_ATTACHED: Interface Vethernet3626
is attached to Network Adapter 8 of gentoo-pxe-520 on port 193 of module
13 with dvport id 6899
2011 Mar 25 11:10:06 n1k-bl %ETHPORT-2-IF_SEQ_ERROR: Error ("Client data
inconsistency") while communicating with component MTS_SAP_ACLMGR for
opcode MTS_OPC_ETHPM_PORT_PRE_CFG (RID_PORT: Vethernet3626)
2011 Mar 25 11:10:06 n1k-bl %ETHPORT-2-IF_DOWN_ERROR_DISABLED: Interface
Vethernet3626 is down (Error disabled. Reason:Client data inconsistency)
```

Example 9-8 show interface brief Command

```
switch# show int brief
--------------------------------------------------------------------------------
Port VRF Status IP Address Speed MTU
--------------------------------------------------------------------------------
management interface0 -- up 172.23.232.141 1000 1500
--------------------------------------------------------------------------------
Ethernet VLAN Type Mode Status Reason Speed Port
Interface Ch #
--------------------------------------------------------------------------------
Eth3/2 1 eth trunk up none 1000(D) --
Eth3/3 1 eth access up none 1000(D) --
switch#
```

Example 9-9 show interface ethernet Command

```
switch# show interface e1/14
e1/7 is down (errDisabled)
```

Example 9-10 show interface ethernet Command

```
switch# show interface eth3/2
Ethernet3/2 is up
  Hardware: Ethernet, address: 0050.5653.6345 (bia 0050.5653.6345)
  MTU 1500 bytes, BW -598629368 Kbit, DLY 10 usec,
     reliability 0/255, txload 0/255, rxload 0/255
  Encapsulation ARPA
  Port mode is trunk
  full-duplex, 1000 Mb/s
  Beacon is turned off
  Auto-Negotiation is turned off
  Input flow-control is off, output flow-control is off
  Auto-mdix is turned on
  Switchport monitor is off
    Rx
    18775 Input Packets 10910 Unicast Packets
    862 Multicast Packets 7003 Broadcast Packets
    2165184 Bytes
    Tx
    6411 Output Packets 6188 Unicast Packets
    216 Multicast Packets 7 Broadcast Packets 58 Flood Packets
    1081277 Bytes
    1000 Input Packet Drops 0 Output Packet Drops
    1 interface resets
switch#
```

Example 9-11 show interface ethernet counters Command

```
switch# show interface eth3/2 counters
--------------------------------------------------------------------------------
Port                InOctets       InUcastPkts      InMcastPkts      InBcastPkts
--------------------------------------------------------------------------------
Eth3/2             2224326         11226           885          7191

--------------------------------------------------------------------------------
Port               OutOctets      OutUcastPkts     OutMcastPkts     OutBcastPkts
--------------------------------------------------------------------------------
Eth3/2             1112171          6368           220             7
```

Example 9-12 show interface vEthernet Command

```
switch# show interface veth1
Vethernet1 is up
    Port description is gentoo1, Network Adapter 1
    Hardware is Virtual, address is 0050.56bd.42f6
    Owner is VM "gentoo1", adapter is Network Adapter 1
    Active on module 33
    VMware DVS port 100
    Port-Profile is vlan48
    Port mode is access
    Rx
    491242 Input Packets 491180 Unicast Packets
    7 Multicast Packets 55 Broadcast Packets
    29488527 Bytes
    Tx
    504958 Output Packets 491181 Unicast Packets
    1 Multicast Packets 13776 Broadcast Packets 941 Flood Packets
    714925076 Bytes
    11 Input Packet Drops 0 Output Packet Drops
switch#
```

Example 9-13 show interface capabilities Command

```
switch# show interface capabilities
management interface0
  Model:                 --
  Type:                  --
  Speed:                 10,100,1000,auto
  Duplex:                half/full/auto
  Trunk encap. type:     802.1Q
  Channel:               no
  Broadcast suppression: none
  Flowcontrol:           rx-(none),tx-(none)
  Rate mode:             none
  QOS scheduling:        rx-(none),tx-(none)
  CoS rewrite:           yes
  ToS rewrite:           yes
  SPAN:                  yes
  UDLD:                  yes
  Link Debounce:         no
  Link Debounce Time:    no
  MDIX:                  no
  Port Group Members:    none

port-channel1
  Model:                 unavailable
  Type:                  unknown
  Speed:                 10,100,1000,10000,auto
  Duplex:                half/full/auto
  Trunk encap. type:     802.1Q
  Channel:               yes
  Broadcast suppression: percentage(0-100)
  Flowcontrol:           rx-(off/on/desired),tx-(off/on/desired)
  Rate mode:             none
  QOS scheduling:        rx-(none),tx-(none)
  CoS rewrite:           yes
  ToS rewrite:           yes
  SPAN:                  yes
  UDLD:                  no
  Link Debounce:         no
  Link Debounce Time:    no
  MDIX:                  no
  Port Group Members:    none

port-channel2
  Model:                 unavailable
  Type:                  unknown
  Speed:                 10,100,1000,10000,auto
  Duplex:                half/full/auto
  Trunk encap. type:     802.1Q
  Channel:               yes
  Broadcast suppression: percentage(0-100)
  Flowcontrol:           rx-(off/on/desired),tx-(off/on/desired)
  Rate mode:             none
  QOS scheduling:        rx-(none),tx-(none)
  CoS rewrite:           yes
  ToS rewrite:           yes
  SPAN:                  yes
  UDLD:                  no
  Link Debounce:         no
  Link Debounce Time:    no
  MDIX:                  no
  Port Group Members:    none

port-channel12
  Model:                 unavailable
  Type:                  unknown
  Speed:                 10,100,1000,10000,auto
  Duplex:                half/full/auto
  Trunk encap. type:     802.1Q
  Channel:               yes
  Broadcast suppression: percentage(0-100)
  Flowcontrol:           rx-(off/on/desired),tx-(off/on/desired)
  Rate mode:             none
  QOS scheduling:        rx-(none),tx-(none)
  CoS rewrite:           yes
  ToS rewrite:           yes
  SPAN:                  yes
  UDLD:                  no
  Link Debounce:         no
  Link Debounce Time:    no
  MDIX:                  no
  Port Group Members:    none

control0
```

```
  Model:                  --
  Type:                   --
  Speed:                  10,100,1000,auto
  Duplex:                 half/full/auto
  Trunk encap. type:      802.1Q
  Channel:                no
  Broadcast suppression: none
  Flowcontrol:            rx-(none),tx-(none)
  Rate mode:              none
  QOS scheduling:         rx-(none),tx-(none)
  CoS rewrite:            yes
  ToS rewrite:            yes
  SPAN:                   yes
  UDLD:                   yes
  Link Debounce:          no
  Link Debounce Time:     no
  MDIX:                   no
  Port Group Members:     none

switch#
```

Example 9-14 show interface virtual port-mapping Command

```
switch# show interface virtual port-mapping

--------------------------------------------------------------------------------
Port    Hypervisor Port    Binding Type    Status    Reason
--------------------------------------------------------------------------------
Veth1   DVPort5747         static          up        none
Veth2   DVPort3361         static          up        none
switch#
```

Example 9-15 module vem execute vemcmd show portsec status Command

```
cyp1-switch# module vem 3 execute vemcmd show portsec status
 LTL if_index Max Aging Aging DSM Sticky VM
 Secure Time Type Bit Enabled Name
 Addresses
 56 1c0000a0 5 0 Absolute Clr No Ostinato-Upgrade-VM1.eth1
```

Example 9-16 show port-security Command

```
switch# show port-security
Total Secured Mac Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 8192

----------------------------------------------------------------------------
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
                (Count)       (Count)          (Count)
----------------------------------------------------------------------------
Vethernet1            1              0              0              Shutdown
==========================================================================
```

Example 9-17 show port-security address interface vethernet Command

```
switch# show port-security address interface vethernet 11
 Secure Mac Address Table
----------------------------------------------------------------------
Vlan/Vxlan Mac Address Type Ports Configured Age
 (mins)
```

```
---------- ----------- ------ ----- ---------------
 50 0050.56a4.38ec STATIC Vethernet11 0
 50 0000.0000.0011 DYNAMIC Vethernet11
```

CHAPTER 10

Port Profiles

This chapter describes how to identify and resolve problems with port profiles and includes the following sections:

Information About Port Profiles

Port profiles are used to configure interfaces. A port profile can be assigned to multiple interfaces tp give them all the same configuration. Changes to the port profile are propagated automatically to the configuration of any interface assigned to it.

In VMware vCenter Server, a port profile is represented as a port group. The vEthernet or Ethernet interfaces are assigned in vCenter Server to a port profile for the following reasons:

- Defining a port configuration by policy.
- Applying a single policy across a large number of ports.
- Supporting both vEthernet and Ethernet ports.

vEthernet port profiles can be assigned by the server administrator to physical ports (a VMNIC or a PNIC). Port profiles not configured as vEthernet can be assigned to a VM virtual port.

Note While a manual interface configuration overrides that of the port profile, we do not recommend that you do so. Manual interface configuration is only used, for example, to quickly test a change or allow a port to be disabled without having to change the inherited port profile.

For more information about assigning port profiles to physical or virtual ports, see your VMware documentation.

To verify that the profiles are assigned as expected to physical or virtual ports, use the following **show** commands:

- **show port-profile virtual usage**
- **show running-config interface** *interface-id*

To verify port profile inheritance, use the following command:

- **show running-config interface** *interface-id*

Note Inherited port profiles cannot be changed or removed from an interface from the Cisco Nexus 1000V CLI. This action can only be done from vCenter Server.

Note Inherited port profiles are automatically configured by the Cisco Nexus 1000V when the ports are attached on the hosts. This action is done by matching up the VMware port group assigned by the system administrator with the port profile that created it.

For detailed information about port profiles, see the *Cisco Nexus 1000V Port Profile Configuration Guide*.

Problems with Port Profiles

The following are symptoms, possible causes, and solutions for problems with port profiles.

Symptom	Possible Causes	Solution
You do not see the port group on vCenter Server or the following message is displayed: `Warning: Operation succeeded locally but update failed on vCenter server. Please check if you are connected to vCenter Server.`	The connection to vCenter server is down.	1. Verify that the connection to vCenter Server is Enabled and Connected. **show svs connections** 2. Reconnect to vCenter server. For detailed instructions, see the *Connecting to vCenter Server* procedure in the *Cisco Nexus 1000V System Management Configuration Guide*.
	The domain configuration was not successfully pushed to vCenter server.	1. Verify that the domain configuration was successfully pushed to vCenter Server. **show svs domain** 2. Fix any problems with the domain configuration. For information about configuring the domain, see the *Cisco Nexus 1000V System Management Configuration Guide*.
	The port profile is configured incorrectly.	1. Verify that the **vmware port-group** is configured for the port profile and that the port profile is enabled. **show port profile name** *name* 2. Fix the port profile using the procedures in the *Cisco Nexus 1000V Port Profile Configuration Guide*.

Symptom	Possible Causes	Solution
A port configuration is not applied to an interface.	Management connectivity between vCenter server and the VSM has prevented the port profile assignment from being sent or received.	1. Display the port profile usage by interface. **show port-profile virtual usage** 2. Verify that the interface level configuration did not overwrite the port profile configuration. **show run** **show port-profile expand-interface** 3. If the **show** command output is incorrect, on vCenter server, reassign the port group to the interface.
An Ethernet interface or vEthernet interface is administratively down. A system message similar to the following is logged: `%VMS-3-DVPG_NICS_MOVED: '1' nics have been moved from port-group 'Access483' to 'Unused_Or_Quarantine_Veth'.`	The interface is inheriting a quarantined port profile. A configuration was not saved prior to rebooting the VSM, the configuration was lost, and the interfaces were moved to one of the following port profiles: • Unused_Or_Quarantine_Uplink for ethernet types • Unused_Or_Quarantine_Veth for Vethernet types	1. Verify the port profile-to-interface mapping. **show port-profile virtual usage** 2. Reassign the VMNIC or PNIC to a non-quarantined port group to enable the interface to be up and forwarding traffic. This requires changing the port group on vCenter Server.
After applying a port profile, an online interface is quarantined. A system message similar to the following is logged: `%PORT-PROFILE-2-INTERFACE_QUARANTINED: Interface Ethernet3/3 has been quarantined due to Cache Overrun`	The assigned port profile is incorrectly configured. The incorrect command fails when the port profile is applied to an interface. Although a specific command fails, the port profile-to-interface mapping is created.	1. Identify the command that failed. **show accounting log | grep FAILURE** 2. Verify that the interface is quarantined. **show port-profile sync-status** 3. Verify the port profile-to-interface mapping. **show port-profile virtual usage** 4. Fix the error in the port profile using the procedures in the *Cisco Nexus 1000V Port Profile Configuration Guide*. 5. Bring the interface out of quarantine. **no shutdown** The interface comes back online. 6. Return shutdown control to the port profile. **default shutdown**

Symptom	Possible Causes	Solution
After modifying a port profile, an assigned offline interface is quarantined. A system message similar to the following is logged: `%PORT-PROFILE-2-INTERFACE_QUARANTINED: Interface Ethernet4/3 has been quarantined due to Cache Overrun`	The interface has been removed from the DVS.	To bring the interface back online, see the "Recovering a Quarantined Offline Interface" section on page 10-4.
A module and all associated interfaces are offline. A system message similar to the following is logged: `2011 Mar 2 22:28:50 switch %VEM_MGR-2-VEM_MGR_REMOVE_NO_HB: Removing VEM 3 (heartbeats lost) 2011 Mar 2 22:29:00 switch %VEM_MGR-2-MOD_OFFLINE: Module 3 is offline`	The interface carrying system VLANs for the module has gone down for one of the following reasons: • System interfaces were removed from the DVS on vCenter Server. • The module was powered down. • There is a general loss of connectivity to the module.	Follow VEM troubleshooting guidelines to bring the module back online To bring the interface back online, see the "Recovering a Quarantined Offline Interface" section on page 10-4.

Recovering a Quarantined Offline Interface

You can recover and bring online an interface that is offline and has been quarantined.

BEFORE YOU BEGIN

- Log in to the CLI in EXEC mode.

DETAILED STEPS

Step 1 Verify that the interface has been quarantined. The interface appears in the **show** command output.

show port-profile sync-status

Step 2 On vCenter server, add or associate the PNIC to a port profile (either the original port profile or a different port profile).

The interface comes back online.

Step 3 Verify that the interface has come back online.

show interface brief

Step 4 Verify the port profile-to-interface mapping.

show port-profile virtual usage

Step 5 Verify the interface has come out of quarantine automatically. The interface should no longer appear in the show command output.

show port-profile sync-status

Step 6 Return shutdown control to the port profile.

default shutdown

Port Profile Logs

To enable and collect detailed logs for port profiles, use the following commands:

- **debug port-profile trace**
- **debug port-profile error**
- **debug port-profile all**
- **debug msp all**

After enabling the debug log, the results of any subsequent port profile configuration are captured in the log file.

Port Profile Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to port profiles.

Command	Purpose
show port-profile	Displays the port profile configuration. See Example 10-1 on page 10-6.
show port-profile name *name*	Displays the configuration for a named port profile. See Example 10-2 on page 10-7.
show port-profile brief	Displays a tabular view of all configured port profiles. See Example 10-3 on page 10-7.
show port-profile expand-interface	Displays all configured port profiles expanded to include the interfaces assigned to them. See Example 10-4 on page 10-7.
show port-profile expand-interface name *name*	Displays a named port profile expanded to include the interfaces assigned to it. See Example 10-5 on page 10-8.
show port-profile-role [**name** *port-profile-role-name*]	Displays the port profile role configuration, including role names, descriptions, assigned users, and assigned groups. See Example 10-7 on page 10-8.
show running-config port-profile [*profile-name*]	Displays the port profile configuration. See Example 10-6 on page 10-8.

Command	Purpose
show port-profile-role	Displays the port profile role configuration. See Example 10-7 on page 10-8.
show port-profile-role users	Displays the available users and groups. See Example 10-8 on page 10-9.
show port-profile sync-status [**interface** *if-name*]	Displays the interfaces that are not synchronized with the port profile. See Example 10-9 on page 10-9.
show port-profile virtual usage [**name** *profile-name*]	Displays the port profile usage by interface. See Example 10-10 on page 10-9.
show msp internal info	Displays the port profile mappings on vCenter server and configured roles. See Example 10-11 on page 10-9.
show system internal port-profile profile-fsm	Displays the port profile activity on the Cisco Nexus 1000V, including transitions such as inherits and configurations. If the following displays, then all inherits are processed: `Curr state: [PPM_PROFILE_ST_SIDLE]` See Example 10-12 on page 10-13.
show system internal port-profile event-history msgs	Displays the messages logged about port profile events within the Cisco Nexus 1000V. See Example 10-13 on page 10-14.

For detailed information about **show** command output, see the *Cisco Nexus 1000V Command Reference*.

EXAMPLES

Example 10-1 show port-profile Command

```
switch# show port-profile
port-profile 1
 type: Vethernet
 description:
 status: enabled
 max-ports: 1
 min-ports: 1
 inherit:
 config attributes:
  switchport mode access
  ip port access-group acl1 in
  capability vxlan
  no shutdown
 evaluated config attributes:
  switchport mode access
  ip port access-group acl1 in
  capability vxlan
  no shutdown
 assigned interfaces:
 port-group: 1
 system vlans: none
 capability l3control: no
 capability iscsi-multipath: no
 capability vxlan: yes
 capability l3-vservice: no
 port-profile role: none
 port-binding: static#
```

Example 10-2 show port-profile name Command

```
switch# show port-profile name vEthProfile3
port-profile 1
 type: Vethernet
 description:
 status: enabled
 max-ports: 1
 min-ports: 1
 inherit:
 config attributes:
  switchport mode access
  ip port access-group acl1 in
  capability vxlan
  no shutdown
 evaluated config attributes:
  switchport mode access
  ip port access-group acl1 in
  capability vxlan
  no shutdown
 assigned interfaces:
 port-group: 1
 system vlans: none
 capability l3control: no
 capability iscsi-multipath: no
 capability vxlan: yes
 capability l3-vservice: no
 port-profile role: none
 port-binding: static
```

Example 10-3 show port-profile brief Command

```
switch# show port-profile brief
VM_PP_NIC8_VLAN_1338          Vethernet  1        3      3      374       0
VM_PP_NIC9_VLAN_1339          Vethernet  1        3      3      374       0
--------------------------------------------------------------------------------
Profile    Assigned  Total  Sys    Parent Child  UsedBy
Type       Intfs     Prfls  Prfls  Prfls  Prfls  Prfls
--------------------------------------------------------------------------------
Vethernet  3549      1524   7      1524   0      18
Ethernet   10        11     4      11     0      8
DAO-VSM#
Vethernet  8
Ethernet   10
switch#
```

Example 10-4 show port-profile expand-interface Command

```
switch# show port-profile expand-interface
port-profile 50
Vethernet6
switchport mode access
switchport access vlan 50
no shutdown
Vethernet27
```

```
switchport mode access
switchport access vlan 50
no shutdown
Vethernet30
switchport mode access
switchport access vlan 50
no shutdown
Vethernet31
switchport mode access
switchport access vlan 50
no shutdown
Vethernet32
switchport mode access
switchport access vlan 50
no shutdownport-profile AccessProf
  id: 1
  capability: 0x0
  state: 0x0
```

Example 10-5 show port-profile expand-interface name Command

```
switch# show port-profile expand-interface name UplinkProfile1
port-profile EthProfile1
Ethernet2/2
    switchport mode trunk
    switchport trunk allowed vlan 110-119
    no shutdown
switch#
```

Example 10-6 show running-config port-profile Command

```
switch# show running-config port-profile
port-profile type ethernet UplinkProfile1
  description "Profile for critical system ports"
  vmware port-group
  switchport mode access
  switchport access vlan 113
  switchport trunk native vlan 113
  channel-group auto mode on
  no shutdown
port-profile type vethernet vEthProfile2
  vmware port-group
  vmware max-ports 5
  switchport mode trunk
  switchport trunk native vlan 112
  channel-group auto mode on sub-group cdp
  no shutdown
switch#
```

Example 10-7 show port-profile-role Command

```
switch# show port-profile-role name adminUser

Name: adminUser
Description: adminOnly
Users:
    hdbaar (user)
Assigned port-profiles:
    allaccess2
switch#
```

Example 10-8 show port-profile-role users Command

```
switch# show port-profile-role users
Groups:
  Administrators
  TestGroupB
Users:
  hdbaar
  fgreen
  suchen
  mariofr
switch#
```

Example 10-9 show port-profile sync-status Command

```
switch# show port-profile sync-status interface ethernet 3/2
Ethernet3/2
 port-profile: uplink
 interface status: quarantine
 sync status: out of sync
 cached commands:
 errors:
    command cache overrun
 recovery steps:
    bring interface online
switch#
```

Example 10-10 show port-profile virtual usage Command

```
switch# show port-profile virtual usage
-------------------------------------------------------------------------------
Port Profile               Port        Adapter        Owner
-------------------------------------------------------------------------------
n1kv-uplink0               Po1
                           Eth3/2      vmnic1         localhost.
                           Eth3/3      vmnic2         localhost.
vlan1767                   Veth7       Net Adapter 1  all-tool-7
                           Veth8       Net Adapter 1  all-tool-8
aipc1765                   Veth4       Net Adapter 1  bl-h-s
inband/outband interface 1766                 Veth6       Net Adapter 3  bl-h-s
mgmt1764                   Veth5       Net Adapter 2  bl-h-s
vpc-mac-uplink             Po7
                           Eth5/2      vmnic1         localhost.
                           Eth5/3      vmnic2         localhost.
ch-vpc-mac-uplink          Po2
                           Po3
                           Eth4/2      vmnic1         VDANIKLNCOS
                           Eth4/3      vmnic2         VDANIKLNCOS
ch-aipc1765                Veth1       Net Adapter 1  bl-h-p
ch-mgmt1764                Veth2       Net Adapter 2  bl-h-p
ch-inband/outband interface1766              Veth3       Net Adapter 3  bl-h-p
switch#
```

Example 10-11 show msp internal info Command

```
switch# show msp internal info
port-profile Access484
      id: 5
      capability: 0x0
      state: 0x1
      type: 0x1
      system vlan mode: -
```

```
  system vlans:
  port-binding: static
  max ports: 256
  vmware config information
    pg name: Access484
    dvs:  (ignore)
  port-profile role:
  alias information:
    pg id: Access484
    dvs uuid:
    type: 1
    pg id: dvportgroup-3285
    dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
    type: 2
    pg id: dvportgroup-3292
    dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
    type: 2
port-profile Unused_Or_Quarantine_Uplink
  id: 1
  capability: 0x1
  state: 0x1
  type: 0x1
  system vlan mode: -
  system vlans:
  port-binding: static
  max ports: 32
  vmware config information
    pg name: Unused_Or_Quarantine_Uplink
    dvs:  (ignore)
  port-profile role:
  alias information:
    pg id: Unused_Or_Quarantine_Uplink
    dvs uuid:
    type: 1
    pg id: dvportgroup-2444
    dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
    type: 2
port-profile Unused_Or_Quarantine_Veth
  id: 2
  capability: 0x0
  state: 0x1
  type: 0x1
  system vlan mode: -
  system vlans:
  port-binding: static
  max ports: 32
  vmware config information
    pg name: Unused_Or_Quarantine_Veth
    dvs:  (ignore)
  port-profile role:
  alias information:
    pg id: Unused_Or_Quarantine_Veth
    dvs uuid:
    type: 1
    pg id: dvportgroup-2445
    dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
    type: 2
port-profile eth-break-deinherit
  id: 10
  capability: 0x1
  state: 0x1
  type: 0x1
  system vlan mode: -
  system vlans:
```

```
  port-binding: static
  max ports: 32
  vmware config information
    pg name: eth-break-deinherit
    dvs:  (ignore)
  port-profile role:
  alias information:
    pg id: eth-break-deinherit
    dvs uuid:
    type: 1
    pg id: dvportgroup-3286
    dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
    type: 2
    pg id: dvportgroup-3293
    dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
    type: 2
port-profile eth-break-inherit
  id: 9
  capability: 0x1
  state: 0x1
  type: 0x1
  system vlan mode: -
  system vlans:
  port-binding: static
  max ports: 32
  vmware config information
    pg name: eth-break-inherit
    dvs:  (ignore)
  port-profile role:
  alias information:
    pg id: eth-break-inherit
    dvs uuid:
    type: 1
    pg id: dvportgroup-3287
    dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
    type: 2
    pg id: dvportgroup-3294
    dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
    type: 2
port-profile uplink
  id: 3
  capability: 0x3
  state: 0x1
  type: 0x1
  system vlan mode: trunk
  system vlans: 480-481
  port-binding: static
  max ports: 32
  vmware config information
    pg name: uplink
    dvs:  (ignore)
  port-profile role:
  alias information:
    pg id: uplink
    dvs uuid:
    type: 1
    pg id: dvportgroup-3283
    dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
    type: 2
port-profile uplink-quar
  id: 12
  capability: 0x1
  state: 0x1
  type: 0x1
```

```
    system vlan mode: -
    system vlans:
    port-binding: static
    max ports: 32
    vmware config information
      pg name: uplink-quar
      dvs:  (ignore)
    port-profile role:
    alias information:
      pg id: uplink-quar
      dvs uuid:
      type: 1
      pg id: dvportgroup-3288
      dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
      type: 2
      pg id: dvportgroup-3295
      dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
      type: 2
  port-profile veth-break-deinherit
    id: 8
    capability: 0x0
    state: 0x1
    type: 0x1
    system vlan mode: -
    system vlans:
    port-binding: static
    max ports: 256
    vmware config information
      pg name: veth-break-deinherit
      dvs:  (ignore)
    port-profile role:
    alias information:
      pg id: veth-break-deinherit
      dvs uuid:
      type: 1
      pg id: dvportgroup-3289
      dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
      type: 2
      pg id: dvportgroup-3296
      dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
      type: 2
  port-profile veth-break-inherit
    id: 7
    capability: 0x0
    state: 0x1
    type: 0x1
    system vlan mode: -
    system vlans:
    port-binding: static
    max ports: 256
    vmware config information
      pg name: veth-break-inherit
      dvs:  (ignore)
    port-profile role:
    alias information:
      pg id: veth-break-inherit
      dvs uuid:
      type: 1
      pg id: dvportgroup-3290
      dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
      type: 2
      pg id: dvportgroup-3297
      dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
      type: 2
  port-profile vpc-uplink
    id: 6
    capability: 0x3
    state: 0x1
    type: 0x1
    system vlan mode: trunk
    system vlans: 480-481
    port-binding: static
    max ports: 32
    vmware config information
      pg name: vpc-uplink
      dvs:  (ignore)
    port-profile role:
    alias information:
      pg id: vpc-uplink
      dvs uuid:
      type: 1
      pg id: dvportgroup-3291
      dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
      type: 2
      pg id: dvportgroup-3298
      dvs uuid: 44 dc 3b 50 53 11 b7 ac-ef ed ef 46 ee df c2 d5
      type: 2
  pending binds:
  port-profile-role adfd
    id: 0
    desc:
    num users: 1
      group GROUP

  switch#
```

Example 10-12 show system internal port-profile profile-fsm Command

```
switch# show system internal port-profile profile-fsm
    >>>>FSM: <PROFILE_FSM:1> has 4 logged transitions<<<<<

    1) FSM:<PROFILE_FSM:1> Transition at 856903 usecs after Tue Mar  8 19:11:47 2011
        Previous state: [PPM_PROFILE_ST_SIDLE]
        Triggered event: [PPM_PROFILE_EV_EIF_STATUS_CHANGE]
        Next state: [PPM_PROFILE_ST_SIDLE]

    2) FSM:<PROFILE_FSM:1> Transition at 858442 usecs after Tue Mar  8 19:11:47 2011
        Previous state: [PPM_PROFILE_ST_SIDLE]
        Triggered event: [PPM_PROFILE_EV_ELEARN]
        Next state: [PPM_PROFILE_ST_SIF_CREATE]

    3) FSM:<PROFILE_FSM:1> Transition at 842710 usecs after Tue Mar  8 19:12:04 2011
        Previous state: [PPM_PROFILE_ST_SIF_CREATE]
        Triggered event: [PPM_PROFILE_EV_EACKNOWLEDGE]
        Next state: [FSM_ST_NO_CHANGE]

    4) FSM:<PROFILE_FSM:1> Transition at 873872 usecs after Tue Mar  8 19:12:04 2011
        Previous state: [PPM_PROFILE_ST_SIF_CREATE]
        Triggered event: [PPM_PROFILE_EV_ESUCCESS]
        Next state: [PPM_PROFILE_ST_SIDLE]

        Curr state: [PPM_PROFILE_ST_SIDLE]
    switch#
```

Example 10-13 show system internal port-profile event-history msgs Command

```
switch# show system internal port-profile event-history msgs
    1) Event:E_MTS_RX, length:60, at 538337 usecs after Tue Mar  8 19:13:02 2011
        [NOT] Opc:MTS_OPC_IM_IF_CREATED(62467), Id:0X0000B814, Ret:SUCCESS
        Src:0x00000101/175, Dst:0x00000101/0, Flags:None
        HA_SEQNO:0X00000000, RRtoken:0x00000000, Sync:UNKNOWN, Payloadsize:120
        Payload:
        0x0000:  00 00 00 02 00 00 00 02 00 00 00 0c 00 00 00 29

    2) Event:E_MTS_RX, length:60, at 515030 usecs after Tue Mar  8 19:13:02 2011
        [NOT] Opc:MTS_OPC_LC_ONLINE(1084), Id:0X0000B7E8, Ret:SUCCESS
        Src:0x00000101/744, Dst:0x00000101/0, Flags:None
        HA_SEQNO:0X00000000, RRtoken:0x00000000, Sync:UNKNOWN, Payloadsize:234
        Payload:
        0x0000:  02 00 00 03 00 00 00 00 00 00 03 02 03 02 00 00

    3) Event:E_MTS_RX, length:60, at 624319 usecs after Tue Mar  8 19:12:05 2011
        [NOT] Opc:MTS_OPC_PPM_INTERFACE_UPDATE(152601), Id:0X00003908, Ret:SUCCESS
        Src:0x00000101/489, Dst:0x00000101/0, Flags:None
        HA_SEQNO:0X00000000, RRtoken:0x00000000, Sync:UNKNOWN, Payloadsize:107
        Payload:
        0x0000:  00 00 00 02 00 00 00 02 00 00 00 0c 00 00 00 26

    4) Event:E_MTS_RX, length:60, at 624180 usecs after Tue Mar  8 19:12:05 2011
        [NOT] Opc:MTS_OPC_PPM_INTERFACE_UPDATE(152601), Id:0X00003905, Ret:SUCCESS
        Src:0x00000101/489, Dst:0x00000101/0, Flags:None
        HA_SEQNO:0X00000000, RRtoken:0x00000000, Sync:UNKNOWN, Payloadsize:107
        Payload:
        0x0000:  00 00 00 02 00 00 00 02 00 00 00 0c 00 00 00 26

    5) Event:E_MTS_RX, length:60, at 624041 usecs after Tue Mar  8 19:12:05 2011
        [NOT] Opc:MTS_OPC_PPM_INTERFACE_UPDATE(152601), Id:0X00003903, Ret:SUCCESS
        Src:0x00000101/489, Dst:0x00000101/0, Flags:None
        HA_SEQNO:0X00000000, RRtoken:0x00000000, Sync:UNKNOWN, Payloadsize:107
        Payload:
        0x0000:  00 00 00 02 00 00 00 02 00 00 00 0c 00 00 00 26
    ...
```

CHAPTER 11

Port Channels and Trunking

This chapter describes how to troubleshoot port channels and trunking and includes the following sections:.

Information About Port Channels and Trunking

This section includes the following topics:

Port Channel Overview

Port channels aggregate multiple physical interfaces into one logical interface to provide higher bandwidth, load balancing, and link redundancy.

A port channel performs the following functions:

- Increases the aggregate bandwidth on a link by distributing traffic among all functional links in the channel.
- Load balances across multiple links and maintains optimum bandwidth usage.
- Provides high availability. If one link fails, traffic previously carried on this link is switched to the remaining links. If a link goes down in a port channel, the upper protocol is not aware of it. To the upper protocol, the link is still there, although the bandwidth is diminished. The MAC address tables are not affected by link failures.

Port Channel Restriction

The following are port channel restrictions.

- Port channels do not support ACLs.
- Port channels do not support NetFlow.

Trunking Overview

Trunking, also known as VLAN trunking, enables interconnected ports to transmit and receive frames in more than one VLAN over the same physical link.

Trunking and port channels function as follows:

- Port channels enable several physical links to be combined into one aggregated logical link.
- Trunking enables a link to carry (trunk) multiple VLAN traffic.

Initial Troubleshooting Checklist

Use the following checklist to begin troubleshooting port channel and trunking issues:

Checklist	√
Use the s**how port-channel compatibility-parameters** CLI command to determine port channel requirements.	
Ensure that all interfaces in the port channel have the same destination device for Link Aggregation Control Protocol (LACP) channels. By using the Asymmetric Port Channel (APC) feature in the Cisco Nexus 1000V, ports in an ON mode channel can be connected to two different destination devices. **Note** APC is supported only on mode channels. It is not supported for LACP channels.	
Verify that either side of a port channel is connected to the same number of interfaces.	
Verify that each interface is connected to the same type of interface on the other side.	
Verify that all required VLANs on a trunk port are in the allowed VLAN list.	
Verify that all the members trying to form a port channel are on the same module.	
Verify that the port channel configuration is present in the profile used by the physical ports.	
Configure APC if the ports are connected to different upstream switches.	
If the upstream switch does not support port channels, make sure that you haveto configure APC in the profile. In addition, make sure that you have no more than two ports in the APC.	

The following commands help you to troubleshoot port channels and trunking:

- **show port-channel summary**
- **show port-channel internal event-history interface port-channel** *channel-number*

- **show port-channel internal event-history interface ethernet** *slot-number*
- **show system internal ethpm event-history interface port-channel** *channel-number*
- **show system internal ethpm event-history interface ethernet** *slot-number*
- **show vlan internal trunk interface ethernet** *slot-number*
- **show vlan internal trunk interface port-channel** *channel-number*
- **debug port-channel error**
- **module vem** *module-number* **execute vemcmd show port**
- **module vem** *module-number* **execute vemcmd show pc**
- **module vem** *module-number* **execute vemcmd show trunk**

Example 11-1 shows output of the **show port-channel summary** command.

Example 11-1 show port-channel summary Command

```
switch# show port-channel summary
Flags:  D - Down        P - Up in port-channel (members)
        I - Individual  H - Hot-standby (LACP only)
        s - Suspended   r - Module-removed
        S - Switched    R - Routed
        U - Up (port-channel)
--------------------------------------------------------------------------------
Group Port-       Type     Protocol  Member Ports
      Channel
--------------------------------------------------------------------------------
1     Po1(SU)     Eth      NONE      Eth3/4(P)
2     Po2(SU)     Eth      NONE      Eth3/2(P)    Eth3/6(P)
```

Troubleshooting Asymmetric Port Channels

When you are troubleshooting asymmetric port channels, follow these guidelines:

- Use APC when you want to configure a port channel whose members are connected to two different upstream switches.
- APC depends on Cisco Discovery Protocol (CDP). Make sure CDP is enabled on the VSM and upstream switches.
- Physical ports within an APC get assigned subgroup IDs based on the CDP information received from upstream switches.
- A user can manually configure subgroup IDs in interface configuration submode.
- Make sure that you configured subgroup CDP either with a port profile or on the port channel interface.
- Ports in APC come up only when they are assigned subgroup IDs manually or through CDP.
- Enter the **show cdp neighbors** command on the VSM and check the output.
- Once the ports came up, check that ports are put in the correct subgroups by entering the **module vem** *module-number* **execute vemcmd show pc** command on the VEM.
- Use the **debug port-channel trace** command to collect information.

Cannot Create Port Channel

Symptom	Possible Cause	Solution
Cannot create a port channel.	The maximum number of port channels has been reached for the system.	Enter the **show port-channel summary** command to verify the number of port channels already configured. You can have a maximum of 256 port channels on the Cisco Nexus 1000V.

Newly Added Interface Does Not Come Online In a Port Channel

Symptom	Possible Cause	Solution
Newly added interface does not come online in a port channel.	The port channel mode is on.	**1.** Make sure that you have the port channel configuration in the port profile (port group) used by that interface. **2.** Check if a port channel is already present on the module that is using the same port profile. If there is, check the running configuration on the port channel and the newly added interface. The interface does not come up if the port channel configurations are different. **3.** If the port channel configuration is different, apply the difference on the newly added interface. Remove the port, and then add it back.
	Interface parameters are not compatible with those of the existing port.	See the "Forcing Port Channel Characteristics onto an Interface" section on page 11-4, to force the physical interface to take on the parameters of the port channel. Use this procedure only if you want to configure the port channel manually and not through the port profile.

Forcing Port Channel Characteristics onto an Interface

You can force the physical interface to take on the characteristics of the port channel. Use this procedure only if you want to configure the port channel manually and not through the port profile.

BEFORE YOUR BEGIN

- Log in to the CLI in configuration mode.
- Make sure that the forced interface has same speed, duplex, and flow control settings as the channel group.

DETAILED STEPS

Step 1 Enter the interface configuration mode.

interface ethernet *slot/port*

You are placed into interface configuration mode.

```
Example:
switch(config)# interface ethernet 1/4
switch(config-if)
```

Step 2 Force the physical interface with an incompatible configuration to join the channel group.

channel-group *channel-number* **force**

The physical interface with an incompatible configuration is forced to join the channel group.

```
Example:
switch(config-if)# channel-group 5 force
switch(config-if)
```

Verifying a Port Channel Configuration

You can debug port channels configured through a port profile.

BEFORE YOUR BEGIN

- Log in to the CLI in configuration mode.

DETAILED STEPS

Step 1 Verify that you have configured a port channel in the profile.

switch# **show port-profile name** *profile-name*

Step 2 Display summary port channel information.

switch# **show port-channel summary**

Step 3 Debug the port channel configuration.

switch# **debug port-channel trace**

VLAN Traffic Does Not Traverse Trunk

Symptom	Possible Cause	Solution
The VLAN traffic does not traverse trunk.	A VLAN is not in the allowed VLAN list.	Add the VLAN to the allowed VLAN list. Use the **switchport trunk allowed vlan add** *vlan-id* command in the profile used by the interface.

CHAPTER 12

Layer 2 Switching

This chapter describes how to identify and resolve problems that relate to Layer 2 switching and includes the following sections:

- Information About Layer 2 Ethernet Switching, page 12-1
- Port Model, page 12-1
- Layer 2 Switching Problems, page 12-4
- Layer 2 Switching Troubleshooting Commands, page 12-7
- Troubleshooting Microsoft NLB Unicast Mode, page 12-12
- Troubleshooting BPDU Guard, page 12-14

Information About Layer 2 Ethernet Switching

The Cisco Nexus1000V is a distributed Layer 2 virtual switch that extends across many virtualized hosts.

It consists of two components:

- The Virtual Supervisor Module (VSM), which is also known as the control plane (CP). The VSM acts as the supervisor and contains the Cisco CLI, configuration, and high-level features.
- The Virtual Ethernet Module (VEM), which is also known as the data plane (DP). The VEM acts as a line card and runs in each virtualized server to handle packet forwarding and other localized functions.

Port Model

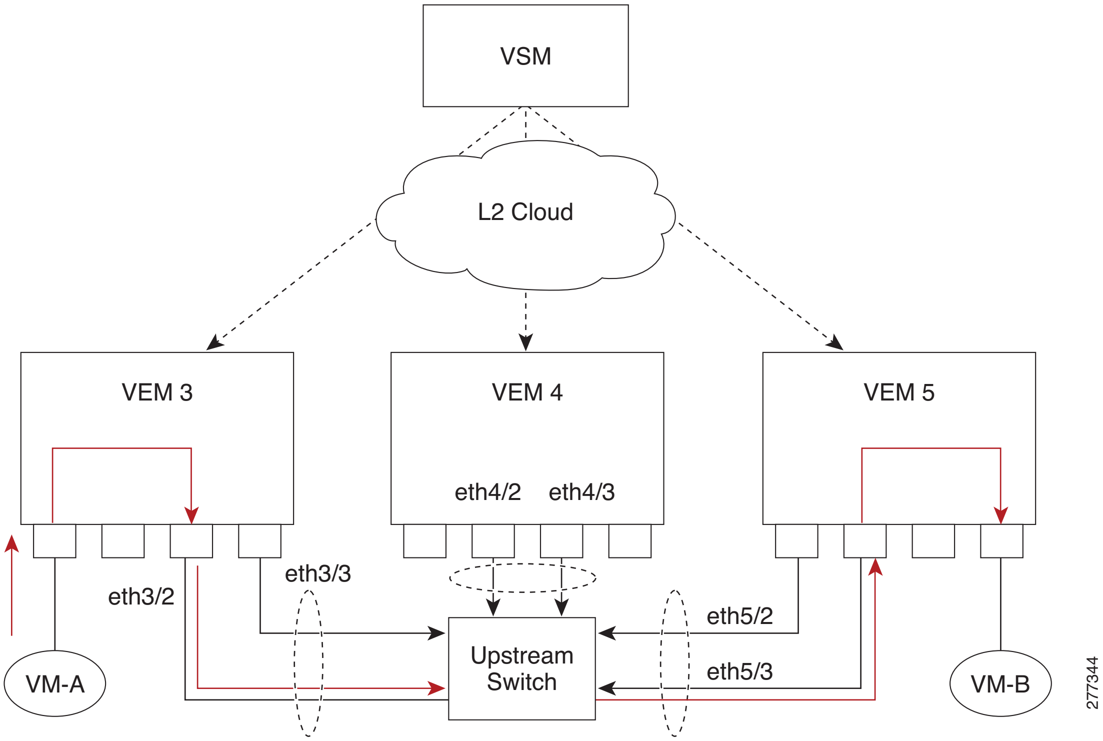
This section includes the following topics:

- Viewing Ports from the VEM, page 12-2
- Viewing Ports from the VSM, page 12-3

Viewing Ports from the VEM

The Cisco Nexus1000V differentiates between virtual and physical ports on each of the VEMs. Figure 12-1 shows how ports on the Cisco Nexus1000V switch are bound to physical and virtual VMware ports within a VEM.

Figure 12-1 VEM View of Ports



On the virtual side of the switch, three layers of ports are mapped together:

- Virtual NICs—Three types of Virtual NICs are in VMware. The virtual NIC (vnic) is part of the VM and represents the physical port of the host that is plugged into the switch. The virtual kernel NIC (VTEP) is used by the hypervisor for management, VMotion, iSCSI, network file system (NFS), and other network access needed by the kernel. This interface carries the IP address of the hypervisor itself and is also bound to a virtual Ethernet port. The vswif (not shown) appears only in CoS-based systems and is used as the VMware management port. Each type maps to a virtual Ethernet port within the Cisco Nexus1000V.
- Virtual Ethernet Ports (VEth)—A vEth port is a port on the Cisco Nexus 1000V. The Cisco Nexus 1000V has a flat space of vEth ports 0..N. The virtual cable plugs into these vEth ports that are moved to the host running the VM.

  Virtual Ethernet ports are assigned to port groups.
- Local Virtual Ethernet Ports (lveth)—Each host has a number of local vEth ports. These ports are dynamically selected for vEth ports that are needed on the host.

  These local ports do not move and are addressable by the module/port number method.

On the physical side of the switch, from bottom to top, is the following:

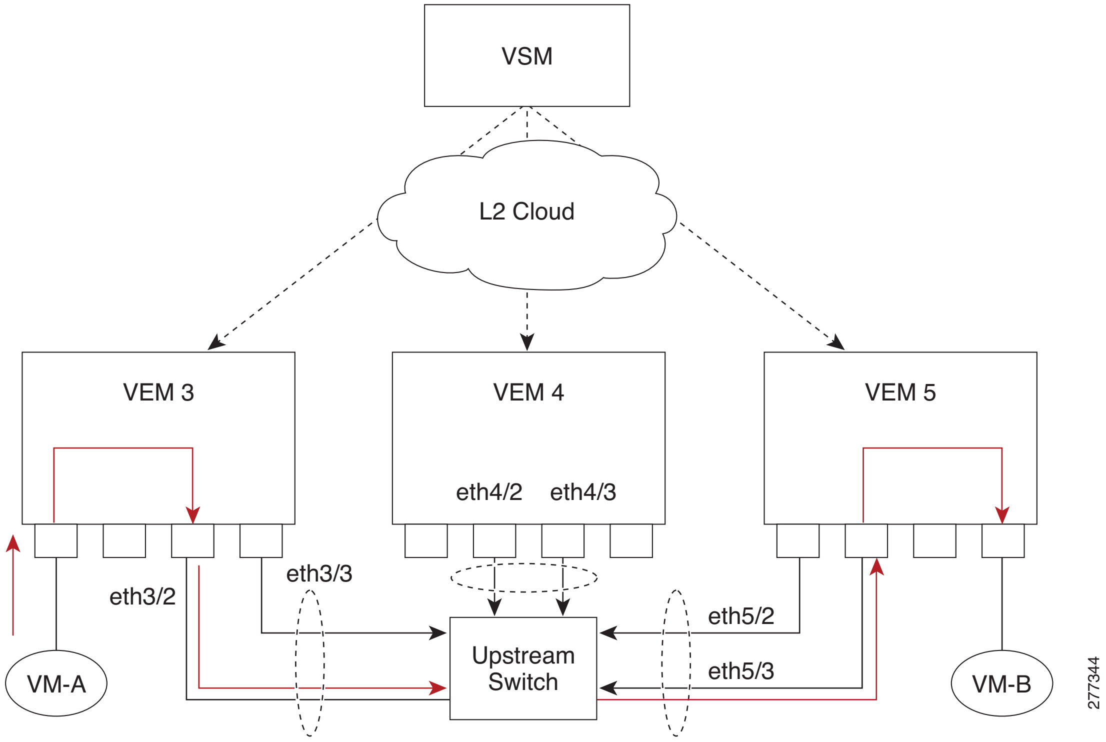
- Each physical NIC in VMware is represented by an interface called a vmnic. The vmnic number is allocated during VMware installation, or when a new physical NIC is installed, and remains the same for the life of the host.
- Each uplink port on the host represents a physical interface. It acts like an lveth port, but because physical ports do not move between hosts, the mapping is 1:1 between an uplink port and a vmnic.
- Each physical port added to the Cisco Nexus1000V switch appears as a physical Ethernet port, just as it would on a hardware-based switch.

The uplink port concept is handled entirely by VMware and is used to associate port configuration with vmnics. There is no fixed relationship between the uplink number and vmnic number. These can be different on different hosts and can change throughout the life of the host. On the VSM, the Ethernet interface number, such as ethernet 2/4, is derived from the vmnic number, not the uplink number.

Viewing Ports from the VSM

Figure 12-2 shows the VSM view ports.

Figure 12-2 VSM View of Ports



Port Types

The following types of ports are available:

- vEths can be associated with any one of the following:
  - VNICs of a Virtual Machine on the ESX host.
  - VTEPs of the ESX Host
  - VSWIFs of an ESX COS Host.
- Eths (physical Ethernet interfaces)—Correspond to the Physical NICs on the ESX host.
- Po (port channel interfaces)—The physical NICs of an ESX Host can be bundled into a logical interface. This logical bundle is referred to as a port channel interface.

For more information about Layer 2 switching, see the *Cisco Nexus 1000V Layer 2 Switching Configuration Guide*.

Layer 2 Switching Problems

This section describes how to troubleshoot Layer 2 problems and lists troubleshooting commands. This section includes the following topics:

- Verifying a Connection Between VEM Ports, page 12-4
- Verifying a Connection Between VEMs, page 12-5
- Isolating Traffic Interruptions, page 12-6

Verifying a Connection Between VEM Ports

You can verify a connection between two vEth ports on a VEM.

Step 1 View the state of the VLANs associated with the port. If the VLAN associated with a port is not active, the port may be down. In this case, you must create the VLAN and activate it.

switch# **show vlan** v*lan-id*

Step 2 View the state of the ports on the VSM.

switch# **show interface brief**

Step 3 Display the ports that are present on the VEM, their local interface indices, VLAN, type (physical or virtual), port mode and port name.

switch# **module vem** *module-number* **execute vemcmd show port**

The key things to look for in the output are as follows:

- State of the port.
- CBL.
- Mode.
- Attached device name.
- The LTL of the port that you are trying to troubleshoot. It will help you to identify the interface quickly in other VEM commands where the interface name is not displayed.

- Make sure that the state of the port is up. If not, verify the configuration of the port on the VSM.

Step 4 View the VLANs and port lists on a particular VEM.

switch# **module vem** *module-number* **execute vemcmd show bd**

If you are trying to verify that a port belongs to a particular VLAN, make sure that you see the port name or LTL in the port list of that VLAN.

Verifying a Connection Between VEMs

You can verify a connection between vEth ports on two separate VEMs.

Step 1 Check if the VLAN associated with the port is created on the VSM.

switch# **show vlan**

Step 2 Check if the ports are up in the VSM.

switch# **show interface brief**

Step 3 On the VEM, check if the CBL state of the two ports is set to the value of 1 for forwarding (active).

switch# **module vem 3 execute vemcmd show port**

Step 4 On the VEM, check if the two vEth ports are listed in the flood list of the VLAN with which they are trying to communicate.

switch# **module vem 3 execute vemcmd show bd**

Step 5 Verify that the uplink switch to which the VEMs are connected is carrying the VLAN to which the ports belong.

Step 6 Find out the port on the upstream switch to which the PNIC (that is supposed to be carrying the VLAN) on the VEM is connected to.

switch# s**how cdp neighbors**

Example:

```
switch# show cdp neighbors

Capability Codes: R - Router, T - Trans-Bridge, B - Source-Route-Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  V - VoIP-Phone, D - Remotely-Managed-Device,
                  s - Supports-STP-Dispute


Device ID              Local Intrfce   Hldtme Capability  Platform      Port ID
swordfish-6k-2         Eth5/2        168    R S I       WS-C6506-E    Gig1/38
```

The PNIC (Eth 5/2) is connected to swordfish-6k-2 on port Gig1/38.

Step 7 Log in to the upstream switch and make sure that the port is configured to allow the VLAN that you are looking for.

```
switch# show running-config interface gigabitEthernet 1/38
Building configuration...

Current configuration : 161 bytes
!
interface GigabitEthernet1/38
 description Srvr-100:vmnic1
 switchport
 switchport trunk allowed vlan 1,60-69,231-233
 switchport mode trunk
end
```

As this output shows, VLANs 1,60-69, 231-233 are allowed on the port. If a particular VLAN is not in the allowed VLAN list, make sure to add it to the allowed VLAN list of the port.

Isolating Traffic Interruptions

You can isolate the cause for no traffic passing across VMs on different VEMs.

Step 1 In the output of the **show port-profile name** command, verify the following information:

- The control and packet VLANs that you configured are present (in the example, these are 3002 and 3003).
- If the physical NIC in your configuration carries the VLAN for the VM, that VLAN is also present in the allowed VLAN list.

```
switch# show port-profile name alluplink
    port-profile alluplink
      description:
      status: enabled
      system vlans: 3002,3003
      port-group: alluplink
      config attributes:
        switchport mode trunk
        switchport trunk allowed vlan 1,80,3002,610,620,630-650
        no shutdown
      evaluated config attributes:
        switchport mode trunk
        switchport trunk allowed vlan 1,80,3002,3003,610,620,630-650
        no shutdown
      assigned interfaces:
        Ethernet2/2
```

Step 2 Inside the VM, verify that the Ethernet interface is up.

ifconfig –a

If not, delete that NIC from the VM, and add another NIC.

Step 3 Using any sniffer tool, verify that ARP requests and responses are received on the VM interface.

Step 4 On the upstream switch, look for the association between the IP and MAC address:

debug arp
show arp

```
Example:
switch# debug arp
ARP packet debugging is on
11w4d: RARP: Rcvd RARP req for 0050.56b7.3031
11w4d: RARP: Rcvd RARP req for 0050.56b7.3031
11w4d: RARP: Rcvd RARP req for 0050.56b7.4d35
11w4d: RARP: Rcvd RARP req for 0050.56b7.52f4
11w4d: IP ARP: rcvd req src 10.78.1.123 0050.564f.3586, dst 10.78.1.24 Vlan3002
11w4d: RARP: Rcvd RARP req for 0050.56b7.3031
switch#

Example:
switch# show arp
```

```
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.78.1.72              -   001a.6464.2008  ARPA
Internet  7.114.1.100             -   0011.bcac.6c00  ARPA   Vlan140
Internet  41.0.0.1                -   0011.bcac.6c00  ARPA   Vlan410
Internet  7.61.5.1                -   0011.bcac.6c00  ARPA   Vlan1161
Internet  10.78.1.5               -   0011.bcac.6c00  ARPA   Vlan3002
Internet  7.70.1.1                -   0011.bcac.6c00  ARPA   Vlan700
Internet  7.70.3.1                -   0011.bcac.6c00  ARPA   Vlan703
Internet  7.70.4.1                -   0011.bcac.6c00  ARPA   Vlan704
Internet  10.78.1.1               0   0011.bc7c.9c0a  ARPA   Vlan3002
Internet  10.78.1.15              0   0050.56b7.52f4  ARPA   Vlan3002
Internet  10.78.1.123            0   0050.564f.3586  ARPA   Vlan3002
```

Step 5 You have completed this procedure.

Layer 2 Switching Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to the Layer 2 MAC address configuration.

Command	Purpose
show mac address-table	Displays the MAC address table to verify all MAC addresses on all VEMs controlled by the VSM. See Example 12-1 on page 12-8.
show mac address-table module *module-number*	Displays all the MAC addresses on the specified VEM.
show mac address-table static *HHHH.WWWW.HHHH*	Displays the MAC address table static entries. See Example 12-2 on page 12-9.
show mac address-table address *HHHH.WWWW.HHHH*	Displays the interface on which the MAC address specified is learned or configured. • For dynamic MAC addresses, if the same MAC address appears on multiple interfaces, each of them is displayed separately. • For static MAC addresses, if the same MAC address appears on multiple interfaces, only the entry on the configured interface is displayed.
show mac address-table static | inc veth	Displays the static MAC address of vEthernet interfaces in case a VEM physical port learns a dynamic MAC addrress and the packet source is in another VEM on the same VSM. See Example 12-3 on page 12-9.
show running-config vlan *vlan-id*	Displays VLAN information in the running configuration.
show vlan [**all-ports | brief | id** *vlan-id* | **name** *name* | **dot1q tag native**]	Displays VLAN information as specified. See Example 12-4 on page 12-9.
show vlan summary	Displays a summary of VLAN information.

Command	Purpose
show interface brief	Displays a table of interface states. See Example 12-5 on page 12-10.
module vem *module-number* **execute vemcmd show port**	On the VEM, displays the port state on a particular VEM. This command can only be used from the VEM. See Example 12-6 on page 12-10.
module vem *module-number* **execute vemcmd show bd**	For the specified VEM, displays its VLANs and their port lists. See Example 12-7 on page 12-11.
module vem *module-number* **execute vemcmd show trunk**	For the specified VEM, displays the VLAN state on a trunk port**.** • If a VLAN is forwarding (active) on a port, its CBL state should be 1. • If a VLAN is blocked, its CBL state is 0. See Example 12-8 on page 12-11.
module vem *module-number* **execute vemcmd show l2** *vlan-id*	For the specified VEM, displays the VLAN forwarding table for a specified VLAN**.** See Example 12-9 on page 12-11.
show interface *interface_id* **mac**	Displays the MAC addresses and the burn-in MAC address for an interface.

Example 12-1 show mac address-table Command

Note The Cisco Nexus 1000V MAC address table does not display multicast MAC addresses.

Tip The “Module” indicates the VEM on which this MAC address is seen.

The “N1KV Internal Port” refers to an internal port created on the VEM. This port is used for control and management of the VEM and is not used for forwarding packets.

```
switch# show mac address-table
VLAN      MAC Address       Type    Age       Port                          Module
---------+-----------------+-------+---------+------------------------------+---------
1         0002.3d11.5502    static  0         N1KV Internal Port            3
1         0002.3d21.5500    static  0         N1KV Internal Port            3
1         0002.3d21.5502    static  0         N1KV Internal Port            3
1         0002.3d31.5502    static  0         N1KV Internal Port            3
1         0002.3d41.5502    static  0         N1KV Internal Port            3
1         0002.3d61.5500    static  0         N1KV Internal Port            3
1         0002.3d61.5502    static  0         N1KV Internal Port            3
1         0002.3d81.5502    static  0         N1KV Internal Port            3
3         12ab.47dd.ff89    static  0         Eth3/3                        3
342       0002.3d41.5502    static  0         N1KV Internal Port            3
342       0050.568d.5a3f    dynamic 0         Eth3/3                        3
343       0002.3d21.5502    static  0         N1KV Internal Port            3
```

```
    343       0050.568d.2aa0    dynamic 9         Eth3/3                         3
    Total MAC Addresses: 13
    switch#
```

Example 12-2 show mac address-table address Command

Tip This command shows all interfaces on which a MAC is learned dynamically.
In this example, the same MAC appears on Eth3/3 and Eth4/3.

```
    switch# show mac address-table address 0050.568d.5a3f
    VLAN      MAC Address       Type    Age       Port                           Module
    ---------+-----------------+-------+---------+------------------------------+---------
    342       0050.568d.5a3f    dynamic 0         Eth3/3                         3
    342       0050.568d.5a3f    dynamic 0         Eth4/3                         4
    Total MAC Addresses: 1
    switch#
```

Example 12-3 show mac address-table static | inc veth Command

```
switch# show mac address-table static | inc veth
460       0050.5678.ed16    static  0         Veth2                          3
460       0050.567b.1864    static  0         Veth1                          4
switch#
```

Example 12-4 show vlan Command

Tip This command shows the state of each VLAN created on the VSM.

```
    switch# show vlan

    VLAN Name                             Status    Ports
    ---- -------------------------------- --------- -------------------------------
    1    default                          active    Eth3/3, Eth3/4, Eth4/2, Eth4/3
    110  VLAN0110                         active
    111  VLAN0111                         active
    112  VLAN0112                         active
    113  VLAN0113                         active
    114  VLAN0114                         active
    115  VLAN0115                         active
    116  VLAN0116                         active
    117  VLAN0117                         active
    118  VLAN0118                         active
    119  VLAN0119                         active
    800  VLAN0800                         active
    801  VLAN0801                         active
    802  VLAN0802                         active
    803  VLAN0803                         active
    804  VLAN0804                         active
    805  VLAN0805                         active
    806  VLAN0806                         active
    807  VLAN0807                         active
    808  VLAN0808                         active
    809  VLAN0809                         active
    810  VLAN0810                         active
    811  VLAN0811                         active
    812  VLAN0812                         active
```

```
813  VLAN0813                         active
814  VLAN0814                         active
815  VLAN0815                         active
816  VLAN0816                         active
817  VLAN0817                         active
818  VLAN0818                         active
819  VLAN0819                         active
820  VLAN0820                         active
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
---- -------------------------------- --------- -------------------------------

Remote SPAN VLANs
-------------------------------------------------------------------------------

Primary  Secondary  Type             Ports
-------  ---------  ---------------  -------------------------------------------
```

Example 12-5 show interface brief Command

```
switch# show interface brief

--------------------------------------------------------------------------------
Port     VRF          Status IP Address                   Speed    MTU
--------------------------------------------------------------------------------
mgmt0     --           up     172.23.232.143               1000     1500

--------------------------------------------------------------------------------
Ethernet      VLAN   Type Mode   Status  Reason                   Speed     Port
Interface                                                                   Ch #
--------------------------------------------------------------------------------
Eth3/4        1      eth  trunk  up      none                       1000(D) --
Eth4/2        1      eth  trunk  up      none                       1000(D) --
Eth4/3        1      eth  trunk  up      none                       1000(D) --
```

Example 12-6 module vem *module-number* execute vemcmd show port Command

Tip Look for the state of the port.

```
~ # module vem 3 execute vemcmd show port
  LTL    IfIndex   Vlan    Bndl  SG_ID Pinned_SGID  Type  Admin State  CBL Mode   Name
    8          0   3969       0      2           2  VIRT     UP    UP    1 Access l20
    9          0   3969       0      2           2  VIRT     UP    UP    1 Access l21
   10          0    115       0      2           0  VIRT     UP    UP    1 Access l22
   11          0   3968       0      2           2  VIRT     UP    UP    1 Access l23
   12          0    116       0      2           0  VIRT     UP    UP    1 Access l24
   13          0      1       0      2           2  VIRT     UP    UP    0 Access l25
   14          0   3967       0      2           2  VIRT     UP    UP    1 Access l26
   16   1a030100      1 T     0      0           2  PHYS     UP    UP    1 Trunk
vmnic1
   17   1a030200      1 T     0      2           2  PHYS     UP    UP    1 Trunk
vmnic2
```

Example 12-7 module vem *module-number* ***execute vemcmd show bd Command***

Tip If a port belongs to a particular VLAN, the port name or LTL should be in the port list for the VLAN.

```
~ # module vem 5 execute vemcmd show bd
Number of valid BDS: 8
BD 1, vdc 1, vlan 1, 2 ports
Portlist:
16 vmnic1
17 vmnic2
BD 100, vdc 1, vlan 100, 0 ports
Portlist:
BD 110, vdc 1, vlan 110, 1 ports
Portlist:
16 vmnic1
BD 111, vdc 1, vlan 111, 1 ports
Portlist:
16 vmnic1
BD 112, vdc 1, vlan 112, 1 ports
Portlist:
16 vmnic1
BD 113, vdc 1, vlan 113, 1 ports
Portlist:
16 vmnic1
BD 114, vdc 1, vlan 114, 1 ports
Portlist:
16 vmnic1
BD 115, vdc 1, vlan 115, 2 ports
Portlist:
10 l22
16 vmnic1
```

Example 12-8 module vem *module-number* ***execute vemcmd show trunk Command***

Tip If a VLAN is active on a port, its CBL state should be 1.
If a VLAN is blocked, its CBL state is 0.

```
~ # module vem 5 execute vemcmd show trunk
Trunk port 16 native_vlan 1 CBL 1
vlan(1) cbl 1, vlan(110) cbl 1, vlan(111) cbl 1, vlan(112) cbl 1, vlan(113) cbl 1,
vlan(114) cbl 1,vlan(115) cbl 1, vlan(116) cbl 1, vlan(117) cbl 1, vlan(118) cbl 1,
vlan(119) cbl 1,
Trunk port 17 native_vlan 1 CBL 0
vlan(1) cbl 1, vlan(117) cbl 1,
~ #
```

Example 12-9 module vem *module-number* ***execute vemcmd show l2 Command***

```
~ # module vem 5 execute vemcmd show l2
Bridge domain 115 brtmax 1024, brtcnt 2, timeout 300
Dynamic MAC 00:50:56:bb:49:d9 LTL 16 timeout 0
Dynamic MAC 00:02:3d:42:e3:03 LTL 10 timeout 0
```

Troubleshooting Microsoft NLB Unicast Mode

Microsoft Network Load Balancing (MS-NLB) is a clustering technology offered by Microsoft as part of the Windows server operating systems. Clustering enables a group of independent servers to be managed as a single system for higher availability, easier manageability, and greater scalability.

For more information about Microsoft Network Load Balancing, see this URL:

http://technet.microsoft.com/en-us/library/bb742455.aspx

Note Access to third-party websites identified in this document is provided solely as a courtesy to customers and others. Cisco Systems, Inc. and its affiliates are not in any way responsible or liable for the functioning of any third-party website, or the download, performance, quality, functioning, or support of any software program or other item accessed through the website, or any damages, repairs, corrections, or costs arising out of any use of the website or any software program or other item accessed through the website. Cisco's End User License Agreement does not apply to the terms and conditions of use of a third-party website or any software program or other item accessed through the website.

Limitations and Restrictions

A syslog is generated if one of the following configurations exists when you try to disable automatic static MAC learning for MS-NLB because they do not support this feature:

- PVLAN port
- Ports configured with unknown unicast flood blocking (UUFB)
- Ports configured with switchport port-security mac-address sticky

Disabling Automatic Static MAC Learning on a vEthernet Interface

You must disable automatic static MAC learning before you can successfully configure NLB on a vEthernet (vEth) interface.

In interface configuration mode enter the following commands:

```
switch(config)# int veth 1
switch(config-if)# no mac auto-static-learn
```

In port profile configuration mode enter the following commands:

```
switch(config)# port-profile type vethernet ms-nlb
switch(config-port-prof)# no mac auto-static-learn
```

Checking Status on a VSM

If the NLB unicast mode configuration does not function, check the status of the Virtual Supervisor Module (VSM).

Confirm that the **no mac auto-static-learn** command is listed in the vEth and/or port profile configurations.

Step 1 In interface configuration mode, generate the VSM status.

```
switch(config-if)# show running-config int veth1
interface Vethernet1
  inherit port-profile vm59
  description Fedora117, Network Adapter 2
  no mac auto-static-learn
  vmware dvport 32 dvswitch uuid "ea 5c 3b 50 cd 00 9f 55-41 a3 2d 61 84 9e 0e c4"
```

Step 2 In port profile configuration mode, generate the VSM status.

```
switch(config-if)# show running-config port-profile ms-nlb
port-profile type vethernet ms-nlb
  vmware port-group
  switchport mode access
  switchport access vlan 59
  no mac auto-static-learn
  no shutdown
  state enabled
```

Checking the Status on a VEM

If the NLB unicast mode configuration does not function, check the status of the Virtual Ethernet Module (VEM). Check the following:

- Confirm that the MS-NLB vEths are disabled.
- Confirm that the MS-NLB shared-MAC (starting with 02:BF) is not listed in the Layer 2 (L2) MAC table.

Step 1 Generate the VEM status.

```
~ # vemcmd show port auto-smac-learning
  LTL   VSM Port  Auto Static MAC Learning
   49      Veth4   DISABLED
   50      Veth5   DISABLED
   51      Veth6   DISABLED
```

Step 2 Generate the Layer 2 MAC address table for VLAN 59.

```
~ # vemcmd show l2 59
Bridge domain   15 brtmax 4096, brtcnt 6, timeout 300
VLAN 59, swbd 59, ""
Flags:  P - PVLAN   S - Secure  D - Drop
       Type         MAC Address   LTL   timeout   Flags    PVLAN
    Dynamic   00:15:5d:b4:d7:02   305         4
    Dynamic   00:15:5d:b4:d7:04   305        25
    Dynamic   00:50:56:b3:00:96    51         4
    Dynamic   00:50:56:b3:00:94   305         5
    Dynamic   00:0b:45:b6:e4:00   305         5
    Dynamic   00:00:5e:00:01:0a    51         0
```

Configuring MS NLB for Multiple VM NICs in the Same Subnet

When MS NLB VMs have more than one port on the same subnet, a request is flooded, which causes both ports to receive it. The server cannot manage this situation.

As a workaround for this situation, enable Unknown Unicast Flood Blocking (UUFB).

Enabling UUFB

To enable UUFB, enter these configuration commands, one on each line. At the end, press **Cntl-Z**.

```
switch# configure terminal
switch (config)# uufb enable
switch (config)#
```

This configuration conceals the requests from the non-NLB ports and allows the system to function as it expected.

Disabling UUFB for VMs That Use Dynamic MAC Addresses

Issues might occur for VMs that use dynamic MAC addresses, other than those MAC addresses assigned by VMware. For ports that host these types of VMs, disable UUFB. To disable UUFB, enter the following commands:

```
switch(config)# int veth3
switch(config-if)# switchport uufb disable
switch(config-if)#
```

Troubleshooting BPDU Guard

BPDU Guard is one of the Spanning Tree Protocol (STP) enhancements. This feature enhances switch network reliability, manageability, and security. It prevent loops and broadcast radiation. We recommend that you enable BPDU guard on access ports so that any end user devices on these ports that have BPDU guard enabled cannot influence the topology. Any malfunctioning device connected to a virtual Ethernet port can flood the Layer 2 network with unwanted BPDUs and causes STP to break down. When you enable BPDU guard on the access-ports, it shuts down the port in the event that it receives a BPDU. To bring up a port disabled by BDPU guard, you must remove the device from the network and then restart the port by entering the **shut/no shut** command.

BPDU Guard Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to the Layer 2 MAC address configuration.

Command	Purpose
show switch edition	Displays the license edition. You must have the ADVANCED 3.0 license for BPDU guard to be enabled in DAOX. See Example 12-10 on page 12-15.
show spanning-tree bpdu guard info	Displays the switch edition and license information. See Example 12-11 on page 12-15.
show run interface *name*	Displays the BPDU guard status on a port profile. See Example 12-12 on page 12-15.

Command	Purpose
show interface virtual spanning-tree bpduguard status	Displays the status or BPDU guard status on vEths. See Example 12-13 on page 12-15.
show system internal cdm info port-profile name *vm*	Displays the status of CDM push for port profile. See Example 12-14 on page 12-15
show system internal cdm info interface *name*	Displays the status of CDM push for a vEth. See Example 12-15 on page 12-16.
vemcmd show card	Displays the global status of BPDU guard on a VEM. See Example 12-16 on page 12-16.
vemcmd show port bpduguard	Displays the status of BPDU guard on a VSM. See Example 12-17 on page 12-16.

Example 12-10 show switch edition Command

```
switch(config)# show switch edition
Switch Edition: ADVANCED (3.0)
Feature Status
Name            State          Licensed    In version
--------------------------------------------------------
bpduguard       enabled         Y           3.0
    Dynamic   00:00:5e:00:01:0a    51         0
```

Example 12-11 show spanning-tree bpduguard info Command

```
switch(config)# show spanning-tree bpduguard info
Global spanning-tree bpduguard status: Enabled
```

Example 12-12 show run interface name Command

```
switch(config-if)# show run interface veth77
interface 77
  inherit port-profile vm
  description fedora20-i386-70, Network Adapter 2
  spanning-tree bpduguard enable
```

Example 12-13 show interface virtual spanning-tree bpduguard status Command

```
switch(config)# show interface virtual spanning-tree bpduguard status
49  Veth36 Enabled
50  Veth68 Enabled
51  Veth73 Enabled
52  Veth77 Enabled
```

Example 12-14 show system internal cdm info port-profile name Command

```
switch(config-if)# show system internal cdm info port-profile name vm
port-profile vm
```

```
  ppid: 4
  eval config:
    spanning-tree bpduguard enable
    no shutdown
    switchport access vlan 59
    switchport mode access
```

Example 12-15 show system internal cdm info interface** name **Command

```
switch(config-if)# show system internal cdm info interface vethernet 77
interface Veth77
  if_index: 0x1c0004a0
  attached: vem 4
  profile: vm (4)
  network: none
  config:
    spanning-tree bpduguard enable
```

Example 12-16 vemcmd show card Command

```
switch# vemcmd show card
Card UUID type  2: 35958c78-bce9-11e0-bd1d-30e4dbc2c276
Card name:
Switch name: switch
…
Licensed: Yes
Global BPDU Guard: Disabled
```

Example 12-17 vemcmd show port bpdugard Command

```
switch# vemcmd show port bpduguard
  LTL   VSM Port  BPDU-Guard
   49     Veth36          -
   50     Veth68          -
   51     Veth73    Enabled
   52     Veth77    Enabled
   53      Veth9    Disabled

   Debugs

vemlogs & DPA logs
Config related:


~ # vemlog debug sfport_orch all
~ # echo "debug sfcdmagent all" > /tmp/dpafifo
~ # echo "debug sfportagent all" > /tmp/dpafifo


Packet path:

# vemlog debug sflayer2 all
~ # echo "debug sfportagent all" > /tmp/dpafifo
```

VLANs

This chapter describes how to identify and resolve problems that might occur when implementing VLANs and includes the following sections:

Information About VLANs

VLANs can isolate devices that are physically connected to the same network but are logically considered to be part of different LANs that do not need to be aware of one another.

We recommend that you use only the following characters in a VLAN name:

- a–z or A–Z
- 0–9
- - (hyphen)
- _ (underscore)

Consider the following guidelines for VLANs:

- Keep user traffic off the management VLAN; keep the management VLAN separate from user data.

Note We recommend that you enable sticky Address Resolution Protocol (ARP) when you configure private VLANs. ARP entries are learned on Layer 3 private VLAN interfaces that are sticky ARP entries. For security reasons, private VLAN port sticky ARP entries do not age out.

- IGMP runs only on the primary VLAN and uses the configuration of the primary VLAN for all secondary VLANs.
- Any IGMP join request in the secondary VLAN is treated as if it is received in the primary VLAN.
- Private VLANs support these Switched Port Analyzer (SPAN) features:
  - You can configure a private VLAN port as a SPAN source port.
  - You can use VLAN-based SPAN (VSPAN) on primary, isolated, and community VLANs or use SPAN on only one VLAN to separately monitor egress or ingress traffic.

- A private VLAN host or promiscuous port cannot be a SPAN destination port. If you configure a SPAN destination port as a private VLAN port, the port becomes inactive.
- A destination SPAN port cannot be an isolated port. (However, a source SPAN port can be an isolated port.)
- SPAN could be configured to span both primary and secondary VLANs or to span either one if the you are is interested only in ingress or egress traffic.
- A MAC address learned in a secondary VLAN is placed in the shared table of the primary VLAN. When the secondary VLAN is associated to the primary VLAN, their MAC address tables are merged into one, shared MAC table.

Initial Troubleshooting Checklist

Troubleshooting a VLAN problem involves gathering information about the configuration and connectivity of individual devices and the entire network. In the case of VLANs, begin your troubleshooting activity as follows:

Checklist	√
Verify the physical connectivity for any problem ports or VLANs.	
Verify that both end devices are in the same VLAN.	

The following CLI commands are used to display VLAN information:

- **show system internal private-vlan info**
- **show system internal private-vlan event-history errors**
- **show system internal private-vlan event-history traces**
- **show vlan id** *vlan-id*
- **show vlan private-vlan**
- **show vlan all-ports**
- **show vlan private-vlan type**
- **show vlan internal bd-info vlan-to-bd 1**
- **show vlan internal errors**
- **show vlan internal info**
- **show vlan internal event-history errors**

Cannot Create a VLAN

Symptom	Possible Cause	Solution
Cannot create a VLAN.	Using a reserved VLAN ID	VLANs 3968 to 4047 and 4094 are reserved for internal use and cannot be changed.

Private VLANs

This chapter describes how to identify and resolve problems related to private VLANs and includes the following sections:

Information About Private VLANs

Private VLANs (PVLANs) are used to segregate Layer 2 Internet service provider (ISP) traffic and convey it to a single router interface. PVLANs achieve device isolation by applying Layer 2 forwarding constraints that allow end devices to share the same IP subnet while being Layer 2 isolated. The use of larger subnets reduces address management overhead. Three separate port designations are used. Each has its own unique set of rules that regulate each connected endpoint's ability to communicate with other connected endpoints within the same private VLAN domain.

Private VLAN Domains

A private VLAN domain consists of one or more pairs of VLANs. The primary VLAN makes up the domain, and each VLAN pair makes up a subdomain. The VLANs in a pair are called the primary VLAN and the secondary VLAN. All VLAN pairs within a private VLAN have the same primary VLAN. The secondary VLAN ID is what differentiates one subdomain from another.

Spanning Multiple Switches

Private VLANs can span multiple switches, just like regular VLANs. Inter-switch link ports do not need to be aware of the special VLAN type and can carry frames tagged with these VLANs as like they do with any other frames. Private VLANs ensure that traffic from an isolated port in one switch does not reach another isolated or community port in a different switch even after traversing an inter-switch link. By embedding the isolation information at the VLAN level and by transporting it along with the packet, you can maintain consistent behavior throughout the network. The mechanism that restricts Layer 2 communication between two isolated ports in the same switch also restricts Layer 2 communication between two isolated ports in two different switches.

Private VLAN Ports

Within a private VLAN domain, there are three separate port designations. Each port designation has its own unique set of rules that regulate the ability of one endpoint to communicate with other connected endpoints within the same private VLAN domain. The following are the three port designations:

- Promiscuous
- Isolated
- Community

For additional information about private VLANs, see the *Cisco Nexus 1000V Layer 2 Switching Configuration Guide*.

Troubleshooting Guidelines

Follow these guidelines when troubleshooting private VLAN issues:

- Use the **show vlan** *private-vlan* command to verify that a private VLAN is configured correctly.
- Use the **show interface** *slot-port* command to verify the interface is up.
- Use the **module vem** *module-number* **execute vemcmd show port** command to verify the VEM is configured correctly.

Private VLAN Troubleshooting Commands

Use the commands listed in this section to troubleshoot problems related to private VLANs.

Command	Purpose
show vlan private-vlan	Displays that a private VLAN is configured correctly. See Example 14-1 on page 14-2.
show interface *name*	Displays that a physical Ethernet interface in a private VLAN trunk promiscuous mode is up. See Example 14-2 on page 14-3.
show interface *veth-name*	Displays that a virtual Ethernet interface in private VLAN host mode is up. See Example 14-3 on page 14-3.
module vem *module-number* **execute vemcmd show port**	Displays that a VEM is configured correctly. See Example 14-4 on page 14-3.

Example 14-1 show vlan private-vlan Command

```
switch# show vlan private-vlan
    Primary  Secondary  Type             Ports
    -------  ---------  ---------------  -------------------------------------------
    152      157        community
    152      158       isolated
    156      153       community
    156      154       community
    156      155       isolated
```

Example 14-2 show interface *name* Command

```
switch# show interface eth3/4
    Ethernet3/4 is up
      Hardware: Ethernet, address: 0050.565a.ca50 (bia 0050.565a.ca50)
      MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
         reliability 0/255, txload 0/255, rxload 0/255
      Encapsulation ARPA
      Port mode is Private-vlan trunk promiscuous
      full-duplex, 1000 Mb/s
      Beacon is turned off
      Auto-Negotiation is turned off
      Input flow-control is off, output flow-control is off
      Auto-mdix is turned on
      Switchport monitor is off
        Rx
        158776 Input Packets 75724 Unicast Packets
        76 Multicast Packets 82976 Broadcast Packets
        13861581 Bytes
        Tx
        75763 Output Packets 75709 Unicast Packets
        3 Multicast Packets 51 Broadcast Packets 0 Flood Packets
        7424670 Bytes
        5507 Input Packet Drops 0 Output Packet Drops
      2 interface resets
```

Example 14-3 show interface *veth* Command

```
switch# show interface v3
    Vethernet3 is up
        Hardware is Virtual, address is 0050.56bb.6330
        Owner is VM "fedora9", adapter is Network Adapter 1
        Active on module 3
        VMware DVS port 10
        Port-Profile is pvlancomm153
        Port mode is Private-vlan host
        Rx
        14802 Input Packets 14539 Unicast Packets
        122 Multicast Packets 141 Broadcast Packets
        1446568 Bytes
        Tx
        15755 Output Packets 14492 Unicast Packets
        0 Multicast Packets 1263 Broadcast Packets 0 Flood Packets
        1494886 Bytes
        45 Input Packet Drops 0 Output Packet Drops
```

Example 14-4 module vem *module-number* execute vemcmd show port Command

```
switch# module vem 3 execute vemcmd show port
      LTL    IfIndex   Vlan    Bndl  SG_ID Pinned_SGID  Type  Admin State  CBL Mode   Name
        8          0   3969       0      2           2  VIRT    UP     UP    4 Access l20
        9          0   3969       0      2           2  VIRT    UP     UP    4 Access l21
       10          0    150       0      2           2  VIRT    UP     UP    4 Access l22
       11          0   3968       0      2           2  VIRT    UP     UP    4 Access l23
       12          0    151       0      2           2  VIRT    UP     UP    4 Access l24
       13          0      1       0      2           2  VIRT    UP     UP    0 Access l25
       14          0   3967       0      2           2  VIRT    UP     UP    4 Access l26
```

```
   16   1a020100      1 T     0      2           2  PHYS     UP    UP     4  Trunk
vmnic1
   18   1a020300      1 T     0      2           2  PHYS     UP    UP     4  Trunk
vmnic3
        pvlan promiscuous trunk port
            153 --> 156
            154 --> 156
            155 --> 156
            157 --> 152
            158 --> 152
   19   1a020400      1 T     0      2           2  PHYS     UP    UP     4  Trunk
vmnic4
        pvlan promiscuous trunk port
            153 --> 156
            154 --> 156
            155 --> 156
            157 --> 152
            158 --> 152
   47   1b020000    154       0      2           0  VIRT     UP    UP     4 Access
fedora9.eth0
        pvlan community 156 153
```

If additional information is required for Cisco Technical Support to troubleshoot a private VLAN issue, use the following commands:

- **show system internal private-vlan info**
- **show system internal private-vlan event-history traces**
- **show system internal private-vlan event-history errors**
- **show system internal private-vlan event-history events**

NetFlow

This chapter describes how to identify and resolve problems that relate to NetFlow and includes the following sections:

- Information About NetFlow, page 15-1
- NetFlow Troubleshooting Commands, page 15-1
- Common NetFlow Problems, page 15-2

Information About NetFlow

NetFlow allows you to evaluate IP traffic and understand how and where it flows. NetFlow gathers data that can be used in accounting, network monitoring, and network planning.

A flow is a one-directional stream of packets that arrives on a source interface (or subinterface), matching a set of criteria. You create a flow using a flow record to define the criteria for your flow. All criteria must match for the packet to count in the given flow. Flows are stored in the NetFlow cache. Flow information tells you the following:

- The source address tells you who is originating the traffic.
- The destination address tells you who is receiving the traffic.
- Ports characterize the application using the traffic.
- Class of service (CoS) examines the priority of the traffic.
- The device interface tells how traffic is being used by the network device.
- Tallied packets and bytes show the amount of traffic.

A flow record defines the information that NetFlow gathers, such as packets in the flow and the types of counters gathered per flow. You can define new flow records or use the predefined Cisco Nexus 1000V flow records.

For detailed information about configuring NetFlow, see the *Cisco Nexus 1000V System Management Configuration Guide*.

NetFlow Troubleshooting Commands

Use the following commands to collect information about NetFlow process runtime configuration errors.

- **show flow internal event-history errors**

Displays event history errors.

- **show flow internal event-history msgs**

  Displays event history messages.

- **show flow internal ddb b**
- **show flow internal mem-stats**

  Displays flow memory statistics to debug memory usage and leaks.

- **debug logfile** *filename*—

  Redirects the output of the following **debug** commands to a file stored in bootflash.

  - **debug nfm all**
  - **vemlog debug sfnetflow_cache all**
  - **vemlog debug sfnetflow_config all**
  - **vemlog debug sfnetflow_flowapi all**

  Enables NetFlow debugging for policy installation on the Virtual Ethernet Module (VEM). Debug messages are printed for every PDL session open, verify, and commit requests that come from the DPA.

- **vemlog debug sfnetflow_flowmon all**
- **vemlog debug sfnetflow_ager all**
- **vemlog debug sfnetflow all**

  Enables packet path debugging for NetFlow policies on the VEM. Debug messages are printed for every packet that hits a NetFlow policy. Use this command with caution. High traffic could result in lot of debug messages.

- **vemcmd show netflow monitor**

  Prints the monitor configuration.

- **vemcmd show netflow interface**

  Prints the interface configuration

- **vemcmd show netflow stats**

  Prints the tracked configuration failures.

The above VEM commands (vemlog and vemcmd) are accessible on the VEM. These commands can be executed from the VSM by preceding them with

module vem *vem-number* **execute**

For example:

VSM command: **module vem 4 execute vemcmd show netflow monitor**

VEM command: **vemcmd show netflow monito**r

- **show flow internal pdl detailed**

  Displays internal flow details.

Common NetFlow Problems

Common NetFlow configuration problems on the VSM can occur if you attempt to do the following:

- Use undefined records, exporters, samplers, or monitors.
- Use invalid records, exporters, samplers, or monitors.
- Modify records, exporters, samplers, or monitors after they are applied to an interface.
- Configure a monitor on an interface that causes the VEM to run out of memory and results in a verification error.
- Use NetFlow in a port channel. NetFlow is not supported in port channels.
- Configure a monitor at multiple levels of a port-profile inheritance tree.

In addition, a configuration error can occur if there is a mismatch between the UDP port configured on the exporter and the port NetFlow Collector has listening turned on. A solution is to provide the version number of the original command to clear the configuration and then reattempt the command.

Debugging a Policy Verification Error

You can debug a policy verification failure due to some processing on the VSM.

Step 1 Enter the **debug nfm all** command.

Step 2 Save the Telnet SSH session buffer to a file.

Step 3 Enter the **ip flow mon** *monitor name direction* command.

The command executes once again and the debug traces are output to the console.

You can also use the policy verification procedure to collect logs for operations such as defining a flow record or tracing exporter functionality.

Debugging Statistics Export

When debugging a NetFlow statistics export problem, follow these guidelines:

- Ensure that the destination IP address is reachable from the VEMs and VSM.
- Ensure that the UDP port configured on the exporter matches that used by the NetFlow Collector.
- View statistics for the exporter and identify any drops by entering the **show flow exporter** command.

CHAPTER 16

ACLs

This chapter describes how to identify and resolve problems that relate to Access Control Lists (ACLs) and includes the following sections:

Information About Access Control Lists

An ACL is an ordered set of rules for filtering traffic. When the device determines that an ACL applies to a packet, it tests the packet against the rules. The first matching rule determines whether the packet is permitted or denied. If there is no match, the device applies a default rule. The device processes packets that are permitted and drops packets that are denied.

ACLs protect networks and specific hosts from unnecessary or unwanted traffic. For example, ACLs are used to disallow HTTP traffic from a high-security network to the Internet. ACLs also allow HTTP traffic but only to specific sites, using the IP address of the site to identify it in an IP ACL.

The following types of ACLs are supported for filtering traffic:

- IP ACLs—The device applies IP ACLs only to IP traffic.
- MAC ACLs—The device applies MAC ACLs only to non-IP traffic.
- IPv6—The device applies IPv6 ACLs only to IPv6 traffic

For detailed information about how ACL rules are used to configure network traffic, see the *Cisco Nexus 1000V Security Configuration Guide*.

ACL Configuration Limits

The following configuration limits apply to ACLs:

- You cannot have more that 128 rules in an ACL.

- The maximum number of ACLs is 128 (spread across all the ACLs) in one VEM.

ACL Restrictions

The following restrictions apply to ACLs:

- You cannot apply more than one IP ACL and one MAC ACL in each direction on an interface.
- A MAC ACL applies only to Layer 2 packets.
- VLAN ACLs are not supported.
- IP fragments are not supported on ACL rules.
- Noninitial fragments are not subject to ACL lookup.
- You cannot have two not-equal-to (neq) operators in the same rule.
- ACL is not supported in port channels.

ACL Troubleshooting Commands

The commands listed in this section can be used on the VSM to see the policies that are configured and applied on the interfaces.

Use the following command to display configured ACLs:

- **show access-list summary**

Use following commands on the VSM to see run-time information of the ACLMGR and ACLCOMP during configuration errors and to collect ACLMGR process run-time information configuration errors:

- **show system internal aclmgr event-history errors**
- **show system internal aclmgr event-history msgs**
- **show system internal aclmgr ppf**
- **show system internal aclmgr mem-stats (to debug memory usage and leaks)**
- **show system internal aclmgr status**
- **show system internal aclmgr dictionary**

Use the following commands to collect ACLCOMP process run-time information configuration errors:

- **show system internal aclcomp event-history errors**
- **show system internal aclcomp event-history msgs**
- **show system internal aclcomp pdl detailed**
- **show system internal aclcomp mem-stats** (to debug memory usage and leaks)

Displaying ACL Policies on the VEM

The commands listed in this section can be used to display configured ACL policies on the Virtual Ethernet Module (VEM).

Use the following command to list the ACLs installed on that server

```
switch(config-if)# module vem 3 execute vemcmd show acl
```

```
AclId RefCnt Type Rules StatId AclName (Stats: Permit/Deny/NoMatch)
----- ------ ---- ----- ------ ------------------------------------
1 0 IPv4 1 1 v4 (Enb: 0/0/0)
2 0 IPv6 0 2 v6 (Dis: 0/0/0)
```

The Acl-id is the local ACLID for this VEM. Ref-cnt refers to the number of instances of this ACL in this VEM.

Use the following command to list the interfaces on which ACLs have been installed

```
~ # module vem 3 execute vemcmd show acl pinst
LTL    Acl-id    Dir
16       1      ingress
```

Debugging Policy Verification Issues

You can debug a policy verification failure.

Note This section is applicable only to VEMs that are available in older releases. The VEMs in the latest release do not have any policy verification failure issue.

Step 1 On the VSM, redirect the output to a file in bootflash.

debug logfile *filename*

Step 2 Enter the **debug aclmgr all** command.

Step 3 Enter the **debug aclcomp all** command.

For the VEMs where the policy exists, or is being applied, enter the following these steps from the VSM. The output goes to the console.

Step 4 Enter the **module vem** *module-number* **execute vemdpalog debug sfaclagent all** command.

Step 5 Enter the **module vem** *module-number* **execute vemdpalog debug sfpdlagent all** command.

Step 6 Enter the **module vem** *module-number* **execute vemlog debug sfacl all** command.

Step 7 Enter the **module vem** *module-number* **execute vemlog start** command.

Step 8 Enter the **module vem** *module-number* **execute vemlog start** command.

Step 9 Configure the policy that was causing the verify error.

Step 10 Enter the **module vem** *module-number* **execute vemdpalog show all** command.

Step 11 Enter **module vem** *module-number* **execute vemlog show all** command.

Save the Telnet or SSH session buffer to a file. Copy the logfile created in bootflash.

Troubleshooting ACL Logging

This section includes the following topics:

- Using the CLI to Troubleshoot ACL Logging on a VEM, page 16-4

- ACL Logging Troubleshooting Scenarios, page 16-5

Using the CLI to Troubleshoot ACL Logging on a VEM

The commands in this section will help you to troubleshoot ACL logging by examining ACL flows.

Viewing Current Flows

You can troubleshoot ACL logging by viewing the current flows on a VEM.

vemcmd show aclflows stats

EXAMPLE

The following example shows how to troubleshoot ACL logging:

```
[root@esx /]# vemcmd show aclflows stats
Current Flow stats:
    Permit Flows:        1647
    Deny Flows:             0
    Current New Flows:    419        --- current new flows yet to be reported.
```

Viewing Active Flows

You can display all the active flows on a VEM.

vemcmd show aclflows [permit | deny]

If you do not specify **permit** or **deny**, the command displays both.

EXAMPLE

The following example shows how to display all the active flows on a VEM:

```
[root@esx /]# vemcmd show aclflows [permit | deny]
If       SrcIP           DstIP           SrcPort DstPort Proto Direction Action   Stats
Veth4    192.168.1.20    192.168.1.10       5345    8080     6 Ingress   permit   1
Veth4    192.168.1.10    192.168.1.20       8080    5769     6 Egress    permit   1
Veth4    192.168.1.20    192.168.1.10       6256    8080     6 Ingress   permit   1
Veth4    192.168.1.10    192.168.1.20       8080    5801     6 Egress    permit   1
Veth4    192.168.1.20    192.168.1.10       5217    8080     6 Ingress   permit   1
Veth4    192.168.1.10    192.168.1.20       8080   57211     6 Egress    permit   1
Veth4    192.168.1.10    192.168.1.20       8080    5865     6 Egress    permit   1
Veth4    192.168.1.10    192.168.1.20       8080    5833     6 Egress    permit   1
Veth4    192.168.1.20    192.168.1.10       5601    8080     6 Ingress   permit   1
Veth4    192.168.1.10    192.168.1.20       8080    5705     6 Egress    permit   1
Veth4    192.168.1.10    192.168.1.20       8080    5737     6 Egress    permit   1
Veth4    192.168.1.20    192.168.1.10       5473    8080     6 Ingress   permit   1
Veth4    192.168.1.20    192.168.1.10      57211    8080     6 Ingress   permit   1
```

Flushing All ACL Flows

You can use the **vemcmd flush aclflows** command to detect any new flows that affect the VEM. Clear all the existing flows, and then you can detect new flows that match any expected traffic. Syslog messages are not sent when you do this action.

Showing Flow Debug Statistics

You can show ACL debug statistics.

To display internal ACL flow statistics, enter the following command:

vemcmd show aclflows dbgstats

To clear all internal ACL flow debug statistics, enter the following command:

vemcmd clear aclflows dbgstats

ACL Logging Troubleshooting Scenarios

This section describes situations that you might encounter when you are using ACL logging.

Troubleshooting a Syslog Server Configuration

If syslog messages are not being sent from the VEM, you can check the syslog server configuration and check if ACL logging is configured by entering the commands shown in the following procedure.

BEFORE YOU BEGIN

- Log in to the VSM and VEM CLI.

PROCEDURE

	Command	Description

Step 1	**`show logging ip access-list status`**
	Example:
	`switch# show logging ip access-list status`
	`switch #`
		Verifies that the remote syslog server is configured properly.

Step 2	**`vemcmd show acllog config`**
	Example:
	`switch# vemcmd show acllog config`
	`switch #`
		Verifies ACL logging on the VEM.

Step 3	**`vemcmd show aclflows dbgstats`**
	Example:
	`switch# vemcmd show aclflows dbgstats`
	`switch #`
		Checks to see if any errors occurred.

Troubleshooting an ACL Rule That Does Not Have a Log Keyword

If the ACL rule does not have a **log** keyword, any flow that matches the ACL is not reported although the ACL statistics continue to advance. You can verify a **log** keyword.

BEFORE YOU BEGIN

- Log in to the VSM and VEM CLI.

PROCEDURE

	Command	Description
Step 1	**show running-config aclmg** **Example** switch# show running-config aclmg switch #	Verifies that the **log** keyword is enabled.
Step 2	**show logging ip access-list status** **Example:** switch# show logging ip access-list status switch #	Verifies that ACL logging is configured properly.
Step 3	**vemcmd show acllog config** **Example:** switch# vemcmd show acllog config switch #	Verifies ACL logging on the VEM.

Troubleshooting a Maximum Flow Limit Value That is Too Low

If the number of flows does not reach 5000 for either permit of deny flows, you can increase the maximum flows.

BEFORE YOU BEGIN

- Log in to the VSM and VEM CLI.

PROCEDURE

	Command	Description
Step 1	**show logging ip access-list status** **Example:** switch# show logging ip access-list status switch #	Verifies that ACL logging is configured properly.

	Command	Description
Step 2	**vemcmd show acllog config** **Example:** switch# vemcmd show acllog config switch #	Verifies ACL logging on the VEM.
Step 3	**logging ip access-list cache max-deny-flows** *<num>* **Example:** switch# logging ip access-list cache max-deny- flows *<num>* switch #	Increases maximum flows to the desired value.

Troubleshooting a Mismatched Configuration Between a VSM and a VEM

If syslog messages are not being sent and the flow information counters are invalid, the configuration between a VSM and a VEM might be mismatched.

Modify any mismatched configurations by using the appropriate configuration command. If the problem persists, enable acllog debugging on both the VSM and the VEM and retry the commands.

BEFORE YOU BEGIN

- Log in to the CLI in EXEC mode.

PROCEDURE

	Command	Description
Step 1	**show logging ip access-list status** **Example:** switch# show logging ip access-list status switch #	Verifies that ACL logging is configured properly.
Step 2	**vemcmd show acllog config** **Example:** switch# vemcmd show acllog config switch #	Verifies ACL logging on the VEM.

CHAPTER 17

Quality of Service

This chapter describes how to identify and resolve problems related to Quality of Service (QoS).

This chapter includes the following sections:

Information About Quality of Service

QoS lets you classify network traffic so that it can be policed and prioritized in a way that prevents congestion. Traffic is processed based on how you classify it and the QoS policies that you put in place. Classification, marking, and policing are the three main features of QoS.

- Traffic Classification—Groups network traffic based on defined criteria.
- Traffic Marking—Modifies traffic attributes such as DSCP, COS, and Precedence by class.
- Policing —Monitors data rates and burst sizes for a particular class of traffic. QoS policing on a network determines whether network traffic is within a specified profile (contract).

For detailed information about QoS, see the *Cisco Nexus 1000V Quality of Service Configuration Guide*.

QoS Configuration Limits

Table 17-1 and Table 17-2 list the configuration limits for QoS.

Table 17-1 QoS Configuration Limits

Item	DVS Limit	Per Server Limit
Class map	1024	1024 (with policies)
Policy map	128	128
Policy instances	12288	1024

Table 17-2 QoS Configuration Limits

Item	Limit
Match criteria per class map	32
Classes per policy map can be of type qos or queuing	64
Match rules under policy map	200

Note We recommend that the class-map should be applied on a maximum of 2000 interfaces. If you apply class maps on more than 2000 interfaces, the **service-policy** command could fail.

QoS Troubleshooting Commands

The commands listed in this section can be used on the VSM to see the policies that are configured and applied on the interfaces.

Command	Purpose
show policy-map [policy-map-name] **show class-map [class-map-name]**	Displays the configured policies and class-maps.
show policy-map interface	Displays the number of packets hitting the configured policies.
show policy-map interface input/output	Displays only the installed policies of type input/output.
show policy-map interface type qos/queuing	Displays the installed policies based on type.
show system internal cdm info app sap 377 detail	Checks the (class map/policy map) configuration delivered by VSM to the connected modules. See Example 17-1 on page 17-3
show resource-availability qos-queuing	Checks whether the QoS configuration is not exceeding the recommended resource limits.
show policy-map interface brief	Displays the installed policies:

Example 17-1 show system internal cdm info app sap 377 detail Command

Contains output logs similar to the following for each class map/policy map:

```
switch# show system internal cdm info app sap 377 detail
policy/377/1/<policy-map/class-map name>
      id: 34
      flags: 0x00000000
      app: Qosmgr SAP (377)
      app_od: 00000af178daed963b0ec2301044c78e81f0cdf70814102aae80a0474ac14...
      app_od_sz: 203
      md5: 16dd64cc7e63fc8681b9357510194fac
```

Use the following commands on the VSM to run-time information for QOSMGR and ACLCOMP during configuration errors.

The commands to collect QOSMGR process run-time information configuration errors are as follows:

- **show system internal ipqos event-history errors**
- **show system internal ipqos event-history msgs**
- **show system internal ipqos mem-stats** (to debug memory usage and leaks)
- **show system internal ipqos status**
- **show system internal ipqos log** (to show aborted plan information)
- **show system internal ipqos**

Troubleshooting the VEM

The commands listed in this section can be used to display configured QoS policies on the VEM.

Command	Purpose
module vem *module-number* **execute vemcmd show qos node**	Lists all class maps and polices in use on the server: See Example 17-2 on page 17-3
module vem *module-number* **execute vemcmd show qos policy**	Lists all the installed policy maps in use on the server. See Example 17-3 on page 17-4
module vem *module-number* **execute vemcmd show qos pinst**	Lists all service policies installed on the server. See Example 17-4 on page 17-4

Example 17-2 module vem module-number execute vemcmd show qos node Command

```
~ # module vem 3 execute vemcmd show qos node
nodeid   type     details
-------- -------- --------
       0   policer
                  cir:50 pir:50
                  bc:200000 be:200000
                  cir/pir units 1 bc/be units 3 flags 2
       1    class   op_AND
                      DSCP
       2    class op_DEFAULT
```

Example 17-3 module vem module-number execute vemcmd show qos policy Command

```
~ # module vem 3 execute vemcmd show qos policy
policyid classid policerid set_type value
-------- -------- -------- -------- --------
       0        1       -1             dscp        5
                 2        0             dscp        0
```

Example 17-4 module vem** module-number **execute vemcmd show qos pinst Command

```
~ # module vem 3 execute vemcmd show qos pinst


id       type
-------- --------
      17  Ingress
            class        bytes matched         pkts matched
         -------- -------------------- --------------------
                1                    0                    0

                2                85529                  572
                0
                  policer stats: conforming (85529, 572)
                  policer stats: exceeding (0, 0)
                  policer stats: violating (0, 0)
```

Debugging Policy Configuration Errors

You can debug a policy configuration failure caused by processing on the VSM.

Step 1 Enter the **debug aclmgr all** command if the policy references an ACL.

Step 2 Enter the **debug ipqos all** command.

Step 3 Enter the policy map and class commands to collect logs for all operations.

Step 4 Save the Telnet SSH session buffer to a file.

If you are debugging a policy on a port profile, it might be easier to first install it directly on an interface.

You can debug a policy configuration failure on the VEM.

Step 1 Enter the **module vem** *module-number* **execute vemdpalog clear** command.

Step 2 Enter the **module vem** *module-number* **execute vemdpalog sfqosagent all** command.

Step 3 Enter **module vem** *module-number* **execute vemdpalog start** command.

Step 4 Enter the **policy-map** command which will execute the command once again with the DPA debug traces output to vemdpalog.

Step 5 Enter **module vem** *module-number* **execute vemdpalog stop** command.

Enter the **module vem** *module-number* **execute vemdpalog show all** command to display the logs on the console.

```
The policy gets added first
add plcy node - calling add policy 8eb5c20 sf_qos_policy_len(policy) 4 classmaps 0, Policy 
name <p_name>
This will be followed by addition of class-map filter nodes. Please note that the same is 
done via multiple sessions. Hence there could be a replace policy , before the addition of 
filter nodes.
A noticeable thing in the log is the class-map counter could be updated.

replace plcy node - calling replace policy 8eb47d8 sf_qos_policy_len(policy) 92 classmaps 
1, Policy <p_name>
…
Adding classmap 1 (108) with op 1 and 2 filters
…
Adding classmap 2 (116) with op 2 and 2 filters
…
Adding classmap 3 (56) with op 0 and 0 filters
…

Every session should end with the log

Debug qosagent: Session commit complete and successful
```

Debugging Policy Verification Failures

You can debug a policy verification failure on VEM.

Step 1 Enter the **module vem module-number execute vemdpalog clear** command.

Step 2 Enter the **module vem module-number execute vemdpalog sfqosagent all** command.

Step 3 Enter the **module vem module-number execute vemdpalog start** command.

Step 4 Enter the **service-policy** command to execute the command once again with the DPA debug traces output to vemdpalog.

Step 5 Enter the **module vem module-number execute vemdpalog stop** command.

Step 6 Enter the **module vem module-number execute vemdpalog show all** command to display the logs on console.

The VEM-side output logs contain the following:

```
add pinst - add_pinst policy_id 0
add pinst - add_pinst gpolicy_id 72352041

verify - installing pinst type 0 49 for policy 0

verify - returned 0

commit - adding pinst ltl 49 use 2 to policy 0

Session commit complete and successful
```

Debugging Policing Configuration Errors

You can debug a policy verification failure caused by processing on the VSM.

Step 1 Enter the **debug aclmgr all** command if the policy references an ACL.

Step 2 Enter the **debug ipqos all** command.

Step 3 Enter the **debug aclcomp all** command.

Step 4 Enter the **service-policy** command to execute the command once again with debug traces output to the console.

Step 5 Save the Telnet SSH session buffer to a file.

If you are debugging a policy on a port profile, it may be easier to first install it directly on an interface.

To debug a policy verification failure on the VEM, follow these steps:

Step 1 Enter the **module vem** *module-number* **execute vemdpalog clear** command.

Step 2 Enter the **module vem** *module-number* **execute vemdpalog sfqosagent all** command.

Step 3 Enter **module vem** *module-number* **execute vemdpalog start** command.

Step 4 Enter the **service-policy** command which will execute the command once again with the DPA debug traces output to vemdpalog.

Step 5 Enter **module vem** *module-number* **execute vemdpalog stop** command.

Step 6 Enter the **module vem** *module-number* **execute vemdpalog show all** command to see the logs on console.

The output will look similar to the following:

```
calling add policy 81610ac len 220 classmaps 3- --> Session actions
…
Adding classmap 1 (108) with op 1 and 2 filters
…
Adding classmap 2 (116) with op 2 and 2 filters
…
Adding classmap 3 (56) with op 0 and 0 filters
…
init pinst ltl 11 policy id 0 if_index 1a020200 --> Service-policy being applied
installing pinst type 0 17 for policy 0
dpa_sf_qos_verify returned 0
…
Session commit complete and successful --> Session ending
```

CHAPTER 18

SPAN

This chapter describes how to identify and resolve problems that relate to SPAN and includes the following topics:

Information About SPAN

The Switched Port Analyzer (SPAN) feature (sometimes called port mirroring or port monitoring) selects network traffic for analysis by a network analyzer. The network analyzer can be a Cisco SwitchProbe or other Remote Monitoring (RMON) probe.

The Cisco Nexus 1000V supports two types of SPAN:

- SPAN (local SPAN) that can monitor sources within a host or VEM.
- Encapsulated remote SPAN (ERSPAN) that can send monitored traffic to an IP destination.

For detailed information about how to configure local SPAN or ERSPAN, see the *Cisco Nexus 1000V System Management Configuration Guide.*

SPAN Session Guidelines

The following are SPAN session guidelines:

- When a SPAN session contains multiple transmit source ports, packets that these ports receive might be replicated even though they are not transmitted on the ports. Examples include the following:
  - Traffic that results from flooding
  - Broadcast and multicast traffic
- For VLAN SPAN sessions with both receive and transmit configured, two packets (one from receive and one from transmit) are forwarded from the destination port if the packets get switched on the same VLAN.
- After VMotion, the following might occur:
  - A session is stopped if the source and destination ports are separated.
  - A session resumes if the source and destination ports end up on the same host.

- The following are required for a running SPAN session:
  - The limit of 64 SPAN sessions is not exceeded.
  - At least one operational source is configured.
  - At least one operational destination is configured.
  - The configured source and destination are on the same host.
  - The session is enabled with the **no shut** command.
- A session is stopped if any of the following occurs:
  - All the source ports go down or are removed.
  - All the destination ports go down or are removed.
  - All the source and destination ports are separated by VMotion.
  - The session is disabled by a **shut** command.

Problems with SPAN

The following are symptoms, possible causes, and solutions for problems with SPAN.

Symptom	Possible Causes	Solution
You observe issues with VM traffic after configuring a session with Ethernet destinations.	—	Ensure that the Ethernet destination is not connected to the same uplink switch. The SPAN packets might cause problems with the IP tables, the MAC tables, or both on the uplink switch, which can cause problems with the regular traffic.
A session state is up and the packets are not received at the destination ports.	—	Verify that the correct VLANs are allowed on the trunk destination ports.
The session displays an error.	—	1. Make sure that VSM-VEM connectivity is working correctly. 2. Force reprogramming of the session on the VEM. **shut** **no shut**
The ERSPAN session is up, but does not see packets at the destination.	The ERSPAN ID is not configured.	Make sure that the ERSPAN ID is configured at the destination.
	An ERSPAN-enabled VMKernel NIC is not configured on the host or VEM.	Make sure that you create a VMKernel NIC on the host using a port profile configured for ERSPAN.
	The ERSPAN-enabled VMKernel NIC is not configured with a proper IP, gateway, or both.	Ping the ERSPAN IP destination from the host VMKernel NIC. **vmkping** *dest-id* Use the **vempkt** command to capture packets on the VMKernel NIC LTL and ensure ERSPAN packets are being sent. Use the **vemlog debug sfspan d** command so that the ERSPAN packets appear in the vempkt capture log.

SPAN Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to SPAN.

Command	Purpose
show monitor	Displays the status of SPAN sessions. See Example 18-1 on page 18-3.
show monitor session	Displays the current state of a SPAN session, the reason it is down, and the session configuration. See Example 18-2 on page 18-3.
module vem *module-number* **execute vemcmd show span**	Displays the VEM source IP and SPAN configuration. See Example 18-3 on page 18-4.

Additional commands:

- **show monitor internal errors**
- **show monitor internal event-history msgs**
- **show monitor internal info global-info**
- **show monitor internal mem-stats**

Example 18-1 show monitor Command

```
switch# show monitor
Session  State        Reason                  Description
-------  -----------  ----------------------  --------------------------------
17       down         Session admin shut      folio
```

Example 18-2 show monitor session Command

```
switch(config)# show monitor session 1
   session 1
---------------
type              : erspan-source
state             : up
source intf       :
    rx            : Eth3/3
    tx            : Eth3/3
    both          : Eth3/3
source VLANs      :
    rx            :
    tx            :
    both          :
filter VLANs      : filter not specified
destination IP    : 10.54.54.1
ERSPAN ID         : 999
ERSPAN TTL        : 64
ERSPAN IP Prec.   : 0
ERSPAN DSCP       : 0
ERSPAN MTU        : 1000
```

Example 18-3 module vem execute vemcmd show span Command

```
switch# vemcmd show span
RX Ltl Sources :52,
TX Ltl Sources :52,
RX Vlan Sources :
TX Vlan Sources :
Source Filter :
2 local 50
RX Ltl Sources :51,
TX Ltl Sources :51,
RX Vlan Sources :
TX Vlan Sources :
Source Filter :
```

CHAPTER 19

Multicast IGMP

This chapter describes how to identify and resolve problems that relate to multicast Internet Group Management Protocol (IGMP) snooping and includes the following sections:

- Information About Multicast, page 19-1
- Problems with Multicast IGMP Snooping, page 19-2

Information About Multicast

IP multicast is a method of forwarding the same set of IP packets to a number of hosts within a network. You can use multicast in an IPv4 network to provide efficient delivery of data to multiple destinations.

Multicast involves both a method of delivery and discovery of senders and receivers of multicast data, which is transmitted on IP multicast addresses called groups. A multicast address that includes a group and source IP address is often referred to as a channel.

Multicast IGMP Snooping

IGMP snooping software examines Layer 2 IP multicast traffic within a VLAN to discover the ports where interested receivers reside. Using the port information, IGMP snooping can reduce bandwidth consumption in a multi-access LAN environment to avoid flooding the entire VLAN. The IGMP snooping feature tracks which ports are attached to multicast-capable routers to help the routers forward IGMP membership reports. The IGMP snooping software responds to topology change notifications.

In general, IGMP snooping works as follows:

- Ethernet switches, such as Cisco Catalyst 6000 Series switches, parse and intercept all IGMP packets and forward them to a CPU, such as a supervisor module, for protocol processing.
- Router ports are learned using IGMP queries. The switch returns IGMP queries, it remembers which port the query comes from, and marks the port as a router port.
- IGMP membership is learned using IGMP reports. The switch parses IGMP report packets and updates its multicast forwarding table to keep track of IGMP membership.
- When the switch receives multicast traffic, it check its multicast table and forwards the traffic only to those ports interested in the traffic.
- IGMP queries are flooded to the whole VLAN.
- IGMP reports are forwarded to the uplink port (the router ports).
- Multicast data traffic is forwarded to uplink ports (the router ports).

Problems with Multicast IGMP Snooping

The operation of multicast IGMP snooping depends on the correct configuration of the upstream switch. Because the IGMP process needs to know which upstream port connects to the router that supports IGMP routing, you must turn on IP multicast routing on the upstream switch by entering the **ip multicast-routing** command.

The following example shows how to turn on global multicast routing, configure an SVI interface, and turn on the PIM routing protocol:

```
switch# conf terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip multicast-routing
switch(config)# end

switch# conf terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# int vlan159
switch(config-if)# ip pim dense-mode
switch(config-if)# end
```

The following example shows a sample Cisco Nexus 5000 Series configuration that has an IGMP querier configured on a VLAN:

```
n5k-sw1(config)# vlan configuration 59
n5k-sw1(config-vlan-config)#   ip igmp snooping querier 7.59.59.1
n5k-sw1(config-vlan-config)#   ip igmp snooping query-interval 60
n5k-sw1(config-vlan-config)#   ip igmp snooping version 3
n5k-sw1(config-vlan-config)#
```

Troubleshooting Guidelines

Follow these guidelines when troubleshooting multicast IGMP issues:

- Use the **show ip igmp snooping** command to verify that IGMP snooping is enabled.
- Make sure that the upstream switch has IGMP configured.
- Use the **show ip igmp snooping groups** command to verify if the Cisco Nexus 1000V switch is configured correctly and is ready to forward multicast traffic. In the displayed output of the command, look for the letter R under the port heading. The R indicates that the Virtual Supervisor Module (VSM) has learned the uplink router port from the IGMP query that was sent by the upstream switch, and means that the Cisco Nexus 1000V is ready to forward multicast traffic.

IGMP Snooping Debugging Commands

You can enable debugging commands for IGMP snooping:

Step 1 Enable logs files on the module that hosts the preferred VMs/Veths.

```
switch(config)# module vem 4 execute vemdpalog debug sfigmp_snoop d
switch(config)# module vem 4 execute vemlog debug sfigmp_snoop d
```

Step 2 (Optional) Clear existing log data.

```
switch(config)# module vem 4 execute vemlog clear
Cleared log
```

Step 3 Start collecting log data.

```
switch(config)# module vem 4 execute vemlog start
Started log
```

Step 4 Wait for the IGMP queries and reports to hit the VEM ports.

Step 5 Stop and verify the log data.

```
switch(config)# module vem 4 execute vemlog stop
Will suspend log after next 0 entries
switch(config)# module vem 4 execute vemlog show all
Timestamp                  Entry CPU  Mod Lv         Message
Jul 15 18:19:27.000679       0  0  99  16   Debug sf_igmp_snoop_thread: IGMP Snoop Thread
waken up
Jul 15 18:19:27.000706        1  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 52
Jul 15 18:19:27.000718        2  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 55
Jul 15 18:19:27.000726        3  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 59
Jul 15 18:19:27.000734        4  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 224.6.7.8, BD: 59
Jul 15 18:19:27.112144        5  2   1  16   Debug IGMP pkt (snoop OFF): orig_src_ltl
0x15, src_ltl 0x40f vlan 232
Jul 15 18:19:27.603386       6  3   1  16   Debug IGMP pkt (snoop ON): orig_src_ltl 0x15,
src_ltl 0x40f vlan 52
Jul 15 18:19:27.603390        7  3   1  16   Debug Notification size: 68
Jul 15 18:19:27.603393        8  3   1  16   Debug Sending IGMP pkt notif: swbd 52,
pkt_size 56, notif_size 68
Jul 15 18:19:27.609442        9  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
IGMP notify message from DP:
Jul 15 18:19:27.609459       10  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
SRC_LTL: 1039, SWBD: 52, pkt_size: 56
Jul 15 18:19:27.609470       11  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
Got IGMP Query.
Jul 15 18:19:27.609479      12  0  99  16   Debug sf_igmp_snoop_handle_query: Received v3
query.
Jul 15 18:19:27.609485       13  0  99  16   Debug sf_igmp_snoop_handle_query: Adding v3
router entry in BD 52 (len: 12).
Jul 15 18:19:27.609494       14  0  99  16   Debug sf_igmp_snoop_add_update_v4_grp:
Existing Group 0.0.0.0 in BD 52.
Jul 15 18:19:27.609502       15  0  99  16   Debug sf_igmp_snoop_add_update_v4_grp:
Existing Member 1039 in Group 0.0.0.0 in BD 52.
Jul 15 18:19:28.011257       16  5   1  16   Debug IGMP pkt (snoop OFF): orig_src_ltl
0x15, src_ltl 0x40f vlan 232
Jul 15 18:19:29.058442       17  0   1  16   Debug IGMP pkt (snoop OFF): orig_src_ltl
0x15, src_ltl 0x40f vlan 180
Jul 15 18:19:30.480455       18  3   1  16   Debug IGMP pkt (snoop OFF): orig_src_ltl
0x15, src_ltl 0x40f vlan 233
Jul 15 18:19:30.623668       19  2   0   0         Started log
Jul 15 18:19:32.002081      20  0  99  16   Debug sf_igmp_snoop_thread: IGMP Snoop Thread
waken up
Jul 15 18:19:32.002103       21  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 52
Jul 15 18:19:32.002111       22  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 55
Jul 15 18:19:32.002117       23  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 59
Jul 15 18:19:32.002122       24  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 224.6.7.8, BD: 59
Jul 15 18:19:34.418381       25 12   1  16   Debug IGMP pkt (snoop ON): orig_src_ltl 0x0,
src_ltl 0x66 vlan 59
Jul 15 18:19:34.418385       26 12   1  16   Debug Notification size: 72
Jul 15 18:19:34.418389       27 12   1  16   Debug Sending IGMP pkt notif: swbd 59,
pkt_size 60, notif_size 72
Jul 15 18:19:34.418400       28 12   1  16   Debug Forward report to router port: 10347
.Jul 15 18:19:34.448932       29  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
IGMP notify message from DP:
Jul 15 18:19:34.448949       30  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
SRC_LTL: 102, SWBD: 59, pkt_size: 60
Jul 15 18:19:34.448961       31  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
Got IGMP v1/v2 Report
Jul 15 18:19:34.448970       32  0  99  16   Debug Handle IGMPv2 report in BD 59, LTL:102,
group: 224.3.4.5.
Jul 15 18:19:34.448978       33  0  99  16   Debug Handle IGMPv2 JOIN in BD 59, LTL:102,
group: 224.3.4.5.
Jul 15 18:19:34.448986       34  0  99  16   Debug sf_igmp_snoop_add_update_v4_grp: Adding
Group 224.3.4.5 to BD 59.
Jul 15 18:19:34.448996       35  0  99  16   Debug sf_igmp_snoop_notify_vsm: Sending to
VSM: opcode : 1, swbd 59, grp_ip: 0xe0030405.
Jul 15 18:19:34.449087       36  0  99  16   Debug sf_igmp_snoop_add_update_v4_grp: Adding
Member 102 to Group 224.3.4.5 in BD 59.
Jul 15 18:19:34.449102       37  0  99  16   Debug sf_igmp_snoop_update_dp: group update
for BD 59: IP: 224.3.4.5, with 2 members
Jul 15 18:19:34.449111       38  0  99  16   Debug sf_igmp_snoop_update_dp: Sending group
update to DP for BD 59: IP: 224.3.4.5, with 2 members
Jul 15 18:19:34.938394       39 14   1  16   Debug IGMP pkt (snoop ON): orig_src_ltl 0x0,
src_ltl 0x66 vlan 59
Jul 15 18:19:34.938400       40 14   1  16   Debug Notification size: 72
Jul 15 18:19:34.938406       41 14   1  16   Debug Sending IGMP pkt notif: swbd 59,
pkt_size 60, notif_size 72
Jul 15 18:19:34.938419       42 14   1  16   Debug Forward report to router port: 10347
.Jul 15 18:19:34.968621       43  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
IGMP notify message from DP:
Jul 15 18:19:34.968634       44  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
SRC_LTL: 102, SWBD: 59, pkt_size: 60
Jul 15 18:19:34.968645       45  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
Got IGMP v1/v2 Report
Jul 15 18:19:34.968654       46  0  99  16   Debug Handle IGMPv2 report in BD 59, LTL:102,
group: 224.3.4.5.
Jul 15 18:19:34.968661       47  0  99  16   Debug Handle IGMPv2 JOIN in BD 59, LTL:102,
group: 224.3.4.5.
Jul 15 18:19:34.968669       48  0  99  16   Debug sf_igmp_snoop_add_update_v4_grp:
Existing Group 224.3.4.5 in BD 59.
Jul 15 18:19:34.968677       49  0  99  16   Debug sf_igmp_snoop_add_update_v4_grp:
Existing Member 102 in Group 224.3.4.5 in BD 59.
Jul 15 18:19:37.000827       50  0  99  16   Debug sf_igmp_snoop_thread: IGMP Snoop Thread
waken up
Jul 15 18:19:37.000853       51  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 52
Jul 15 18:19:37.000895       52  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 55
Jul 15 18:19:37.000905       53  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 59
Jul 15 18:19:37.000912       54  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 224.3.4.5, BD: 59
Jul 15 18:19:37.000919       55  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 224.6.7.8, BD: 59
Jul 15 18:19:37.085327       56  8   1  16   Debug IGMP pkt (snoop ON): orig_src_ltl 0x0,
src_ltl 0x66 vlan 59
Jul 15 18:19:37.085331       57  8   1  16   Debug Notification size: 72
Jul 15 18:19:37.085335       58  8   1  16   Debug Sending IGMP pkt notif: swbd 59,
pkt_size 60, notif_size 72
Jul 15 18:19:37.085345       59  8   1  16   Debug Forward report to router port: 10347
.Jul 15 18:19:37.085998       60  1   1  16   Debug IGMP pkt (snoop ON): orig_src_ltl
0x15, src_ltl 0x40f vlan 59
Jul 15 18:19:37.086002       61  1   1  16   Debug Notification size: 68
Jul 15 18:19:37.086006       62  1   1  16   Debug Sending IGMP pkt notif: swbd 59,
pkt_size 56, notif_size 68
Jul 15 18:19:37.134375       63  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
IGMP notify message from DP:
Jul 15 18:19:37.134390       64  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
SRC_LTL: 102, SWBD: 59, pkt_size: 60
Jul 15 18:19:37.134400       65  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
Got IGMP v1/v2 Report
Jul 15 18:19:37.134409       66  0  99  16   Debug Handle IGMPv2 report in BD 59, LTL:102,
group: 224.3.4.5.
Jul 15 18:19:37.134416       67  0  99  16   Debug Handle IGMPv2 LEAVE in BD 59, LTL:102,
group: 224.3.4.5.
Jul 15 18:19:37.134439       68  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
IGMP notify message from DP:
Jul 15 18:19:37.134446       69  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
SRC_LTL: 1039, SWBD: 59, pkt_size: 56
Jul 15 18:19:37.134453       70  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
Got IGMP Query.
Jul 15 18:19:37.134461       71  0  99  16   Debug sf_igmp_snoop_handle_query: Received v2
query.
Jul 15 18:19:37.134467       72  0  99  16   Debug sf_igmp_snoop_handle_query: Got group
specific query for 0x50403e0.
Jul 15 18:19:37.134475       73  0  99  16   Debug sf_igmp_snoop_start_leave_timers: Found
group 0xe0030405.
Jul 15 18:19:37.134482       74  1   1  16   Debug IGMP pkt (snoop ON): orig_src_ltl 0x15,
src_ltl 0x40f vlan 59
Jul 15 18:19:37.134486       75  1   1  16   Debug Notification size: 68
Jul 15 18:19:37.134488       76  1   1  16   Debug Sending IGMP pkt notif: swbd 59,
pkt_size 56, notif_size 68
Jul 15 18:19:37.134483       77  0  99  16   Debug sf_igmp_snoop_start_leave_timers: Start
leave timer on member 102 for 2 secs.
Jul 15 18:19:37.134504       78  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
IGMP notify message from DP:
Jul 15 18:19:37.134511       79  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
SRC_LTL: 1039, SWBD: 59, pkt_size: 56
Jul 15 18:19:37.134518       80  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
Got IGMP Query.
Jul 15 18:19:37.134524       81  0  99  16   Debug sf_igmp_snoop_handle_query: Received v2
query.
Jul 15 18:19:37.134530       82  0  99  16   Debug sf_igmp_snoop_handle_query: Got group
specific query for 0x50403e0.
Jul 15 18:19:37.134536       83  0  99  16   Debug sf_igmp_snoop_start_leave_timers: Found
group 0xe0030405.
Jul 15 18:19:37.610484       84  5   1  16   Debug IGMP pkt (snoop ON): orig_src_ltl 0x15,
src_ltl 0x40f vlan 52
Jul 15 18:19:37.610489       85  5   1  16   Debug Notification size: 68
Jul 15 18:19:37.610492       86  5   1  16   Debug Sending IGMP pkt notif: swbd 52,
pkt_size 56, notif_size 68
Jul 15 18:19:37.648380       87  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
IGMP notify message from DP:
Jul 15 18:19:37.648396       88  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
SRC_LTL: 1039, SWBD: 52, pkt_size: 56
Jul 15 18:19:37.648406       89  0  99  16   Debug sf_igmp_snoop_v4_pkt_notify_handler:
Got IGMP Query.
Jul 15 18:19:37.648415       90  0  99  16   Debug sf_igmp_snoop_handle_query: Received v3
query.
Jul 15 18:19:37.648422       91  0  99  16   Debug sf_igmp_snoop_handle_query: Adding v3
router entry in BD 52 (len: 12).
Jul 15 18:19:37.648431       92  0  99  16   Debug sf_igmp_snoop_add_update_v4_grp:
Existing Group 0.0.0.0 in BD 52.
Jul 15 18:19:37.648439       93  0  99  16   Debug sf_igmp_snoop_add_update_v4_grp:
Existing Member 1039 in Group 0.0.0.0 in BD 52.
Jul 15 18:19:42.002071       94  0  99  16   Debug sf_igmp_snoop_thread: IGMP Snoop Thread
waken up
```

```
Jul 15 18:19:42.002099       95  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 52
Jul 15 18:19:42.002112       96  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 55
Jul 15 18:19:42.002121       97  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 0.0.0.0, BD: 59
Jul 15 18:19:42.002128       98  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 224.3.4.5, BD: 59
Jul 15 18:19:42.002135       99  0  99  16   Debug sf_igmp_snoop_thread: Check timed-out
members in 224.6.7.8, BD: 59
Jul 15 18:19:43.301931      100  6   0   0         Suspending log
switch(config)#
```

Multicast IGMP Snooping Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to multicast IGMP snooping.

- **show cdp neighbor**

  Displays if IGMP uses the packet VLAN to forward IGMP packets to the VSM, which is the same mechanism that CDP uses. However, if you have disabled the CDP protocol on the upstream switch using the **no cdp enable** command, the **show cdp neighbor** command will not display any information.

Example 19-1 show cdp neighbor Command

```
switch# show cdp neighbor
Capability Codes: R - Router, T - Trans-Bridge, B - Source-Route-Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  V - VoIP-Phone, D - Remotely-Managed-Device,
                  s - Supports-STP-Dispute

Device ID              Local Intrfce   Hldtme  Capability  Platform    Port ID
switch         Eth3/2            179    R S I       WS-C6506-E    Gig5/16
switch         Eth3/4            179    R S I       WS-C6506-E    Gig5/23
```

- **show ip igmp groups**

  Displays whether IGMP snooping is enabled on the VLAN.

Example 19-2 show ip igmp snooping vlan Command

```
switch# show ip igmp snooping vlan 159
IGMP Snooping information for vlan 159
  IGMP snooping enabled     <-- IGMP SNOOPING is enabled for vlan 159
  IGMP querier none
  Switch-querier disabled
  IGMPv3 Explicit tracking enabled (initializing, time-left: 00:03:20)
  IGMPv2 Fast leave disabled
  IGMPv1/v2 Report suppression enabled
  IGMPv3 Report suppression disabled
  Router port detection using PIM Hellos, IGMP Queries
  Number of router-ports: 0
  Number of groups: 0
```

- **show ip igmp snooping groups**

```
switch# show ip igmp snooping groups vlan 1784
```

```
Type: S - Static, D - Dynamic, R - Router port

Vlan  Group Address      Ver  Type  Port list
1784  */*                -    R     Po1  Po2  Eth5/31
1784  227.0.0.1          v2   D     Veth79 Veth80

VSM-DAO# show ip igmp snooping querier vlan 1784
Vlan  IP Address       Version   Expires     Port
1784  184.184.0.12     v3        00:04:14    Po1
1784  184.184.0.12     v3        00:04:14    Po2
1784  184.184.0.12     v3        00:04:14    Eth5/31

switch# show ip igmp snooping groups vlan 1784 detail
IGMP Snooping group membership for vlan 1784
  Group addr: 227.0.0.1
    Group ver: v2 [old-host-timer: not running]
  report-timer: not-running
    Last reporter: 184.184.0.11
    IGMPv1/v2 memb ports:
      Veth79 [0 GQ missed]
      Veth80 [0 GQ missed]


switch# show ip igmp snooping groups vlan 1784 summary
Legend: E - Enabled, D - Disabled

Vlan Snoop (*,G)-Count
1784  E      2
Total number of (*,G) entries: 2
switch#
```

Example 19-3 debug ip igmp snooping vlan Command

```
switch(config)# debug ip igmp snooping vlan
2014 Jul 8 23:49:16.633077 igmp[3157]: SNOOP: Switchport interface Veth43 (308) has been
created, obtaining any static mrouter/oif configs
2014 Jul 8 23:49:16.683929 igmp[3157]: SNOOP: Switchport interface Veth37 (128) has been
created, obtaining any static mrouter/oif configs
2014 Jul 8 23:49:16.748355 igmp[3157]: SNOOP: <vlan 1> clear port:Veth43, vlan:1
2014 Jul 8 23:49:16.789832 igmp[3157]: SNOOP: Switchport interface Veth47 (428) has been
created, obtaining any static mrouter/oif configs
2014 Jul 8 23:49:16.797079 igmp[3157]: SNOOP: Switchport interface Veth38 (158) has been
created, obtaining any static mrouter/oif configs
2014 Jul 8 23:49:16.824702 igmp[3157]: SNOOP: <vlan 11> Added Veth43 to active ports for
vlan 11
2014 Jul 8 23:49:16.824854 igmp[3157]: SNOOP: Mode for if(Vethernet43): 0x80000 vlan: 11
2014 Jul 8 23:49:16.862531 igmp[3157]: SNOOP: <vlan 1> clear port:Veth37, vlan:1
2014 Jul 8 23:49:16.950490 igmp[3157]: SNOOP: <vlan 11> Added Veth37 to active ports for
vlan 11
2014 Jul 8 23:49:16.950638 igmp[3157]: SNOOP: Mode for if(Vethernet37): 0x80000 vlan: 11
2014 Jul 8 23:49:16.998800 igmp[3157]: SNOOP: <vlan 1> clear port:Veth38, vlan:1
2014 Jul 8 23:49:16.999030 igmp[3157]: SNOOP: <vlan 1> clear port:Veth47, vlan:1
2014 Jul 8 23:49:17.089056 igmp[3157]: SNOOP: Switchport interface Veth40 (218) has been
created, obtaining any static mrouter/oif configs
2014 Jul 8 23:49:17.121007 igmp[3157]: SNOOP: Switchport interface Veth39 (188) has been
created, obtaining any static mrouter/oif configs
2014 Jul 8 23:49:17.131549 igmp[3157]: SNOOP: <vlan 11> Added Veth38 to active ports for
vlan 11
2014 Jul 8 23:49:17.131693 igmp[3157]: SNOOP: Mode for if(Vethernet38): 0x80000 vlan: 11
2014 Jul 8 23:49:17.156004 igmp[3157]: SNOOP: <vlan 11> Added Veth47 to active ports for
vlan 11
2014 J
```

Note Even if you enable the **debug** command for IGMP snooping, log details are not available for multicast groups and their members.

Example 19-4 module vem *module-number execute* ***vemcmd show vlan Command***

```
switch# module vem 3 execute vemcmd show vlan 159
BD 159, vdc 1, vlan 159, 3 ports
Portlist:
     18  vmnic3
     47  fedora8.eth0

Multicast Group Table:
Group 224.1.2.3 RID 1 Multicast LTL 4408
    47
    18
Group 0.0.0.0 RID 2 Multicast LTL 4407
    18
```

On the VSM, use the following command:

- **module vem 3 execute vemcmd show igmp 1784**

  In Example 19-2, global IGMP snooping is enabled on VLAN 1784 (the disabled global state takes precedence)

  Multicast group table values are as follows:

  Group 227.0.0.1, Multicast LTL: 10363

  Group */*, Multicast LTL: 10358

- **module vem 3 execute vemcmd show igmp 1784 de**

  In Example 19-2, global IGMP snooping is enabled on VLAN 1784 (the disabled global state takes precedence)

  Multicast group table values are as follows:

  Group 227.0.0.1, Multicast LTL: 10363

  Members: 59, 1039

  Group */*, Multicast LTL: 10358

  Members: 1039

  Querier Info -

  IP Address: 184.184.0.12

  Uptime: 241955 seconds

  Version: 3

  Timeout: 8 seconds

- **module vem** *module-number* **execute vemcmd show vlan**

  In Example 19-4, the output shows that LTL 18 corresponds to vmnic3, and LTL 47 corresponds to VM fedora8, interface eth0.

  The multicast group table for 224.1.2.3 shows the interfaces that the VEM forwards to when it receives multicast traffic for group 224.1.2.3. If fedora8 has multicast group 224.1.2.3 on its eth0 interface, LTL 47 should be in the multicast group table for 224.1.2.3.

LTL 18 is also in multicast group 224.1.2.3, which means it is a VM and generates multicast traffic to 224.1.2.3. The traffic is forwarded to vmnic3, which is the uplink to the upstream switch.

The multicast group table entry for 0.0.0.0 serves as a default route. If any multicast group traffic does not match any of the multicast group, the address uses the default route, which means that the traffic is forwarded to an upstream switch through vmnic3.

Problems with Multicast IGMP Snooping

The following are symptoms, possible causes, and solutions for problems with multicast IGMP snooping.

Symptom	Possible Causes	Solution
A VM is interested in the multicast traffic but is not receiving the multicast traffic.	—	Use the **debug ip igmp snooping vlan** command to determine if IGMP snooping is working as expected. Examine the output to see if the port is receiving the IGMP report and if the interface has been added to the multicast traffic interface list for the VM.
	—	Use the **module vem** *module-number* **execute vemcmd show vlan** command to verify that the multicast distribution table in the VEM has the correct information in it.
	—	Use the **module vem** *module-number* **execute vemcmd show port** command to see the port table. Make sure that the table has the correct information in it. Make sure that the state of the trunk port and the access port is UP/UP.

CHAPTER 20

DHCP, DAI, and IPSG

This chapter describes how to identify and resolve problems related to the following security features:

- Dynamic Host Configuration Protocol (DHCP) snooping
- Dynamic ARP Inspection (DAI)
- IP Source Guard (IPSG)

This chapter includes the following sections:

Information About DHCP Snooping

DHCP snooping acts like a firewall between untrusted hosts and trusted DHCP servers by doing the following:

- Validates DHCP messages received from untrusted sources and filters out invalid response messages from DHCP servers.
- Builds and maintains the DHCP snooping binding database, which contains information about untrusted hosts with leased IP addresses.
- Uses the DHCP snooping binding database to validate subsequent requests from untrusted hosts.

Dynamic ARP inspection (DAI) and IP Source Guard also use information stored in the DHCP snooping binding database.

For detailed information about configuring DHCP snooping, see the *Cisco Nexus 1000V Security Configuration Guide*.

Information About Dynamic ARP Inspection

Dynamic ARP Instpection (DAI) is used to validate ARP requests and responses as follows:

- Intercepts all ARP requests and responses on untrusted ports.
- Verifies that a packet has a valid IP-to-MAC address binding before updating the ARP cache or forwarding the packet.
- Drops invalid ARP packets.

DAI can determine the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a Dynamic Host Configuration Protocol (DHCP) snooping binding database. This database is built by DHCP snooping when it is enabled on the VLANs and on the device. It might also contain static entries that you have created.

For detailed information about configuring DAI, see the *Cisco Nexus 1000V Security Configuration Guide*.

Information About IP Source Guard

IP Source Guard is a per-interface traffic filter that permits IP traffic only when the IP address and MAC address of each packet matches the IP and MAC address bindings of dynamic or static IP source entries in the Dynamic Host Configuration Protocol (DHCP) snooping binding table.

For detailed information about configuring IP Source Guard, see the *Cisco Nexus 1000V Security Configuration Guide*.

Guidelines and Limitations for Troubleshooting

The following guidelines and limitations apply when troubleshooting DHCP snooping, Dynamic ARP Inspection, or IP Source Guard:

- A maximum of 12,000 DHCP entries can be snooped and learned system-wide in the DVS. This combined total is for both entries learned dynamically and entries configured statically.
- Rate limits on interfaces must be set to high values for trusted interfaces such as VSD SVM ports or vEthernet ports that connect to DHCP servers.
- Rate limits for trusted interfaces will be ignored.
- A maximum of 2000 DHCP entries per host can be learned dynamically and configured statically.
- A maximum of 1000 static DHCP entries per interface can be configured.

For detailed guidelines and limitations used in configuring these features, see the *Cisco Nexus 1000V Security Configuration Guide*.

Problems with DHCP Snooping

The following are symptoms, possible causes, and solutions for problems with DHCP snooping.

Symptom	Possible Causes	Solution
With snooping configured, the DHCP client is not able to obtain an IP address from the server.	The IP address was not added to the binding database. A faulty connection is between the DHCP server and client.	1. Verify the connection between the DHCP server(s) and the host connected to the client. **vmkping** 2. If the connection between the DHCP server and the host is broken, do the following: – Check the configuration in the upstream switch, for example, verifying that the VLAN is allowed. – Make sure that the server is up and running.
	The interface of the DHCP server(s) connected to the DVS as a VM is not trusted.	1. On the Virtual Supervisor Module (**VSM**), verify that the interface is trusted. **show ip dhcp snooping** 2. On the VSM, verify that the vEthernet interface attached to the server is trusted. **module vem** *mod#* **execute vemcmd show dhcps interfaces**
	DHCP requests from the VM are not reaching the server for acknowledgement.	On the DHCP server, log in and use a packet capture utility to verify requests and acknowledgements in packets.
	DHCP requests and acknowledgements are not reaching the Cisco Nexus 1000V.	• From the client vEthernet interface, SPAN the packets to verify they are reaching the client. • On the host connected to the client, enable VEM packet capture to verify incoming requests and acknowledgements in packets.
	The Cisco Nexus 1000V is dropping packets.	On the VSM, verify DHCP statistics. **show ip dhcp snooping statistics** **module vem** *mod#* **execute vemcmd show dhcps stats**

Troubleshooting Dropped ARP Responses

The following are possible causes, and solutions for dropped ARP responses.

Possible Causes	Solution
ARP inspection is not configured on the VSM	On the VSM, verify that ARP inspection is configured as expected. **show ip arp inspection** For detailed information about configuring DAI, see the *Cisco Nexus 1000V Security Configuration Guide*.
DHCP snooping is not enabled globally on the VSM or is not enabled on the VLAN.	On the VSM, verify the DHCP snooping configuration. **show ip dhcp snooping** For detailed information about enabling DHCP and configuring DAI, see the *Cisco Nexus 1000V Security Configuration Guide*.
DHCP snooping is not enabled on the VEM or is not enabled on the VLAN.	1. From the VSM, verify the VEM DHCP snooping configuration. **module vem** *mod#* **execute vemcmd show dhcps vlan** 2. Do one of the following: – Correct any errors in the VSM DHCP configuration. For detailed information, see the *Cisco Nexus 1000V Security Configuration Guide*. – If the configuration appears correct on the VSM but fails on the VEM, capture and analyze the error logs from both the VSM and the VEM to identify the reason for the failure.
If snooping is disabled, the binding entry is not statically configured in the binding table.	1. On the VSM, display the binding table. **show ip dhcp snooping binding** 2. Correct any errors in the static binding table. For detailed information about clearing entries from the table, enabling DHCP, and configuring DAI, see the *Cisco Nexus 1000V Security Configuration Guide*.
The binding that corresponds to the VM sending the ARP response is not present in the binding table.	1. On the VSM, display the binding table. **show ip dhcp snooping binding** 2. Correct any errors in the static binding table. For detailed information about clearing entries from the table, enabling DHCP, and configuring DAI, see the *Cisco Nexus 1000V Security Configuration Guide*. 3. If all configurations are correct, make sure to turn on DHCP snooping before DAI or IPSG to make sure the Cisco Nexus 1000V has enough time to add the binding in the snooping database. For more information, see the *Cisco Nexus 1000V Security Configuration Guide*.

Problems with IP Source Guard

The following are symptoms, possible causes, and solutions for problems with IP Source Guard.

Symptom	Possible Causes	Solution
Traffic disruptions	ARP inspection is not configured on the VSM.	On the VSM, verify that IP Source Guard is configured as expected. **show port-profile name** *profile_name* **show running interface** *if_ID* **show ip verify source** For detailed information about configuring IP Source Guard, see the *Cisco Nexus 1000V Security Configuration Guide*
	The IP address that corresponds to the vEthernet interface is not in the snooping binding table.	1. On the VSM, display the binding table. **show ip dhcp snooping binding** 2. Configure the missing static entry or renew the lease on the VM. 3. On the VSM, display the binding table again to verify that the entry is added correctly. **show ip dhcp snooping binding**

Collecting and Evaluating Logs

You can use the commands in this section from the VSM to collect and view logs related to DHCP, DAI, and IP Source Guard.

- VSM Logging, page 20-5
- Host Logging, page 20-6

VSM Logging

You can use the commands in this section from the VSM to collect and view logs related to DHCP, DAI, and IP Source Guard.

VSM Command	Description
debug dhcp all	Enables debug all for dhcp configuration flags
debug dhcp cdm-errors	Enables debugging of cdm errors
debug dhcp cdm-events	Enables debugging of cdm events
debug dhcp errors	Enables debugging of errors
debug dhcp mts-errors	Enables debugging of mts errors
debug dhcp mts-events	Enables debugging of mts events
debug dhcp pkt-events	Enables debugging of pkt events

VSM Command	Description
debug dhcp pss-errors	Enables debugging of pss errors
debug dhcp pss-events	Enables debugging of pss events

Host Logging

You can use the commands in this section from the ESX host to collect and view logs related to DHCP, DAI, and IP Source Guard.

ESX Host Command	Description
echo "logfile enable" > /tmp/dpafifo	Enables DPA debug logging. Logs are output to /var/log/vemdpa.log file.
echo "debug sfdhcpsagent all" > /tmp/dpafifo	Enables DPA DHCP agent debug logging. Logs are output to /var/log/vemdpa.log file.
vemlog debug sfdhcps all	Enables data path debug logging, and captures logs for the data packets sent between the client and the server.
vemlog debug sfdhcps_pod all	Captures POD (Port Opaque Data) logging for the feature.
vemlog debug sfdhcps_config all	Enables data path debug logging, and captures logs for configuration coming from the VSM.
vemlog debug sfdhcps_binding_table all	Enables data path debug logging, and captures logs that correspond to binding database changes.

DHCP, DAI, and IPSG Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to DHCP snooping, DAI, and IP Source Guard.

Command	Description
show running-config dhcp	Displays the DHCP snooping, DAI, and IP Source Guard configuration See Example 20-1 on page 20-7.
show ip dhcp snooping	Displays general information about DHCP snooping. See Example 20-2 on page 20-7.
show ip dhcp snooping binding	Displays the contents of the DHCP snooping binding table. See Example 20-3 on page 20-8.

Command	Description
show feature	Displays the features available, such as DHCP, and whether they are enabled. See Example 20-4 on page 20-8.
show ip arp inspection	Displays the status of DAI. See Example 20-5 on page 20-8.
show ip arp inspection interface vethernet *interface-number*	Displays the trust state and ARP packet rate for a specific interface. See Example 20-6 on page 20-8.
show ip arp inspection vlan *vlan-ID*	Displays the DAI configuration for a specific VLAN. See Example 20-7 on page 20-9.
show ip verify source	Displays interfaces where IP source guard is enabled and the IP-MAC address bindings. See Example 20-8 on page 20-9.
show system internal dhcp {*event-history* | *mem-stats* | *msgs*}	Debugs any issues in the filter-mode configuration. See Example 20-9 on page 20-9, Example 20-10 on page 20-9, and Example 20-11 on page 20-10.
debug dhcp all	Enables debug all for DHCP configuration flags on the VSM. See Example 20-12 on page 20-10.

Example 20-1 show running-config dhcp Command

```
switch# show running-config dhcp

!Command: show running-config dhcp
!Time: Wed Feb 16 14:20:36 2011

version 4.2(1)SV1(4)
feature dhcp

no ip dhcp relay

switch#
```

Example 20-2 show ip dhcp snooping Command

```
switch# show ip dhcp snooping
DHCP snooping service is enabled
Switch DHCP snooping is enabled
DHCP snooping is configured on the following VLANs:
1,13
DHCP snooping is operational on the following VLANs:
1
Insertion of Option 82 is disabled
Verification of MAC address is enabled
DHCP snooping trust is configured on the following interfaces:
Interface              Trusted
------------           -------
vEthernet 3            Yes
```

```
switch#
```

Example 20-3 show ip dhcp snooping binding Command

```
switch# show ip dhcp snooping binding
MacAddress         IpAddress        LeaseSec  Type        VLAN  Interface
-----------------  ---------------  --------  ----------  ----  -------------
0f:00:60:b3:23:33  10.3.2.2         infinite  static      13    vEthernet 6
0f:00:60:b3:23:35  10.2.2.2         infinite  static      100   vEthernet 10
switch#
```

Example 20-4 show feature Command

```
switch# show feature
Feature Name          Instance  State
--------------------  --------  --------
dhcp-snooping         1         enabled
http-server           1         enabled
ippool                1         enabled
lacp                  1         enabled
lisp                  1         enabled
lisphelper            1         enabled
netflow               1         disabled
port-profile-roles    1         enabled
private-vlan          1         disabled
sshServer             1         enabled
tacacs                1         enabled
telnetServer          1         enabled
switch#
```

Example 20-5 show ip arp inspection Command

```
cyp1-switch(config)# show ip arp inspection

Source Mac Validation : Disabled
Destination Mac Validation : Disabled
IP Address Validation : Disabled

Filter Mode(for static bindings): IP-MAC

Vlan : 1
-----------
Configuration : Disabled
Operation State : Inactive

Vlan : 40
-----------
Configuration : Disabled
Operation State : Inactive
```

Example 20-6 show ip arp inspection interface vethernet Command

```
switch# show ip arp inspection interface vethernet 6

 Interface        Trust State
 -------------    -----------
 vEthernet 6        Trusted
switch#
```

Example 20-7 show ip arp inspection vlan Command

```
switch# show ip arp inspection vlan 13

Source Mac Validation      : Disabled
Destination Mac Validation : Enabled
IP Address Validation      : Enabled

switch#
```

Example 20-8 show ip verify source Command

```
cyp1-switch# show ip verify source
Filter Mode(for static bindings): IP-MAC
IP source guard is enabled on the following interfaces:
------------------------------------------------------
Vethernet11


IP source guard operational entries:
-----------------------------------
Interface Filter-mode IP-address Mac-address Vlan
------------ ----------- ---------- -------------- ----
Vethernet11 active 205.2.5.80 00:50:56:a4:38:ec 5
```

Example 20-9 show system internal dhcp *event-history msgs* ***Command***

```
switch# show system internal dhcp event-history msgs
1) Event:E_MTS_RX, length:60, at 809122 usecs after Mon Oct  8 20:59:08 2012
    [RSP] Opc:MTS_OPC_PDL32(148511), Id:0X00F132AB, Ret:SUCCESS
    Src:0x00000302/747, Dst:0x00000201/360, Flags:None
    HA_SEQNO:0X00000000, RRtoken:0x00009498, Sync:UNKNOWN, Payloadsize:132
    Payload:
    0x0000:  00 00 00 03 00 00 00 01 00 00 00 64 00 00 00 07

2) Event:E_MTS_RX, length:60, at 809100 usecs after Mon Oct  8 20:59:08 2012
    [RSP] Opc:MTS_OPC_PDL32(148511), Id:0X00E01555, Ret:SUCCESS
    Src:0x00000502/747, Dst:0x00000201/360, Flags:None
    HA_SEQNO:0X00000000, RRtoken:0x00009497, Sync:UNKNOWN, Payloadsize:132
    Payload:
    0x0000:  00 00 00 03 00 00 00 01 00 00 00 64 00 00 00 07

3) Event:E_MTS_RX, length:60, at 809079 usecs after Mon Oct  8 20:59:08 2012
    [RSP] Opc:MTS_OPC_PDL32(148511), Id:0X006BE1FC, Ret:SUCCESS
    Src:0x00000602/747, Dst:0x00000201/360, Flags:None
    HA_SEQNO:0X00000000, RRtoken:0x00009496, Sync:UNKNOWN, Payloadsize:132
    Payload:
    0x0000:  00 00 00 03 00 00 00 01 00 00 00 64 00 00 00 07

4) Event:E_MTS_RX, length:60, at 809028 usecs after Mon Oct  8 20:59:08 2012
    [RSP] Opc:MTS_OPC_PDL32(148511), Id:0X00F132AA, Ret:SUCCESS
    Src:0x00000302/747, Dst:0x00000201/360, Flags:None
    HA_SEQNO:0X00000000, RRtoken:0x00009474, Sync:UNKNOWN, Payloadsize:132
    Payload:
    0x0000:  00 00 00 03 00 00 00 01 00 00 00 64 00 00 00 07
contd.
```

Example 20-10 show system internal dhcp *mem-stats detail* ***Command***

```
VSM-N1k# show system internal dhcp mem-stats detail
```

```
Private Mem stats for UUID : Malloc track Library(103) Max types: 5
--------------------------------------------------------------------------------
TYPE NAME                                            ALLOCS                BYTES
                                               CURR     MAX       CURR       MAX
   2 MT_MEM_mtrack_hdl                           33      34      19236     19384
   3 MT_MEM_mtrack_info                         588     880       9408     14080
   4 MT_MEM_mtrack_lib_name                     882    1174      42246     56230
--------------------------------------------------------------------------------
Total bytes: 70890 (69k)
--------------------------------------------------------------------------------

Private Mem stats for UUID : Non mtrack users(0) Max types: 149
--------------------------------------------------------------------------------
TYPE NAME                                            ALLOCS                BYTES
                                               CURR     MAX       CURR       MAX
  11 [r-xp]/isan/plugin/0/isan/lib/libavl.so   3421    3421      68360     68360
  26 [r-xp]/isan/plugin/0/isan/lib/libddbcom    116     141     302445    308307
  47 [r-xp]/isan/plugin/0/isan/lib/libindxob      6       6        456       456
  50 [r-xp]/isan/plugin/0/isan/lib/libip.so.      1       1        212       212
  64 [r-xp]/isan/plugin/0/isan/lib/libmpmts.      0       9          0       785
  66 [r-xp]/isan/plugin/0/isan/lib/libmts.so     10      11        972       984
  68 [r-xp]/isan/plugin/0/isan/lib/libnetsta      1       2        704      1350
  81 [r-xp]/isan/plugin/0/isan/lib/libpss.so    158     262     101579    204281
  85 [r-xp]/isan/plugin/0/isan/lib/libsdb.so     44      44       3914      3914
  89 [r-xp]/isan/plugin/0/isan/lib/libsmm.so      3       3        216       216
 111 [r-xp]/isan/plugin/0/isan/lib/libutils.      4       7         69       349
 112 [r-xp]/isan/plugin/0/isan/lib/libvdc_mg      0       1          0        20
 118 [r-xp]/isan/plugin/2/isan/bin/dhcp_snoo      0       2          0        64
 121 [r-xp]/isan/plugin/2/isan/lib/libpdlser      4      29        208      1016
 128 [r-xp]/lib/ld-2.3.3.so                      33      33       5363      5371
 131 [r-xp]/lib/tls/libc-2.3.3.so                51      51       1347      1637
 134 [r-xp]/lib/tls/libpthread-2.3.3.so           1       1         33        33
 138 [r-xp]/usr/lib/libglib-2.0.so.0.600.1       15      16      10372     10392
 145 [r-xp]/isan/plugin/1/isan/lib/libvem_mg      0       1          0      1940
--------------------------------------------------------------------------------
Total bytes: 496250 (484k)
--------------------------------------------------------------------------------
contd.
```

Example 20-11 show system internal dhcp msgs Command

```
switch# show system internal dhcp msgs
1) Event:E_DEBUG, length:75, at 409832 usecs after Mon Oct  8 20:57:48 2012
    [16843009] Session close, handle -767541913, sess-id 0xff0101ba02812d08, state 3

2) Event:E_DEBUG, length:62, at 399944 usecs after Mon Oct  8 20:57:48 2012
    [16843009] PPF session open session-id 0xff0101ba02812d08, msg_id 0

3) Event:E_DEBUG, length:30, at 399866 usecs after Mon Oct  8 20:57:48 2012
    [16843009] PPF goto setting state 1

4) Event:E_DEBUG, length:23, at 682346 usecs after Mon Oct  8 20:57:11 2012
    [16843009] Processed log-mts
contd
```

Example 20-12 debug dhcp all Command

```
switch# debug dhcp all
#
```

Storm Control

This chapter describes how to identify and resolve the problems related to Storm control.

This chapter includes the following sections:

- Information About Storm Control, page 21-1
- Troubleshooting Storm Control, page 21-1

Information About Storm Control

You can use the traffic storm control feature to prevent disruptions from a broadcast, multicast, or unknown-unicast traffic storm.

Troubleshooting Storm Control

This section describes the different types of troubleshooting commands to debug Storm Control:

- Troubleshooting VSM Commands, page 21-1
- Troubleshooting VEM Commands, page 21-1
- Debugging Storm Control on a VEM, page 21-2

Troubleshooting VSM Commands

Displays the detailed storm control statistics on an interface:

- **show storm-control statistics interface** *interface-type module-number/port-number*
- **show storm-control statistics module** *module-number*

Troubleshooting VEM Commands

Displays all the statistics related to broadcast, multcast and unknown unicast traffic:

- **vemcmd show storm stats**

Displays the configured storm rate on a Virtual Ethernet Module (VEM):

- **vemcmd show storm-rate ltl <ltl>**

Displays the storm control status of whether the port is dropping or allowing traffic on a VEM.

- **vemcmd show storm status**

Debugging Storm Control on a VEM

You can debug storm control on a VEM.

Step 1 vemlog clear.

Step 2 vemlog start.

Step 3 vemlog debug sfstormcontrol all.

Step 4 vemlog show all.

CHAPTER 22

System

This chapter describes how to identify and resolve problems related to the Nexus 1000V system.

This chapter includes the following sections:

Information About the System

Cisco Nexus 1000V provides Layer 2 switching functions in a virtualized server environment. Nexus 1000V replaces virtual switches within ESX servers and allows users to configure and monitor the virtual switch using the Cisco NX-OS command line interface. Nexus 1000V also gives you visibility into the networking components of the ESX servers and access to the virtual switches within the network.

The Nexus 1000V manages a data center defined by the vCenter Server. Each server in the Datacenter is represented as a linecard in Nexus 1000V and can be managed as if it were a line card in a physical Cisco switch. The Nexus 1000V implementation has two components:

- Virtual supervisor module (VSM) – This is the control software of the Nexus 1000V distributed virtual switch. It runs on a virtual machine (VM) and is based on NX-OS.
- Virtual Ethernet module (VEM) – This is the part of Cisco Nexus 1000V that actually switches data traffic. It runs on a VMware ESX 4.0 host. Several VEMs are controlled by one VSM. All the VEMs that form a switch domain should be in the same virtual Datacenter as defined by VMware vCenter Server.

See the *Cisco Nexus 1000V Getting Started Guide* for a detailed overview of how the Nexus 1000V works with VMware ESX software.

General Restrictions for vCenter Server

When you are troubleshooting issues related to vCenter Server, make sure that you observe the following restrictions:

- The name of a distributed virtual switch (DVS) name must be unique across Datacenters
- You create a DVS in a network folder
- A Datacenter cannot be removed unless the DVS folder or the underlying DVS is deleted.
- A DVS can be deleted only with the help of VSM using the **no vmware dvs** command in config-svs-conn mode.
- The no vmware dvs command can succeed only if there are no VMs using the DVS port-groups.
- A port group on vCenter Server can be deleted only if there are no interfaces associated with it.
- A sync operation performed in conjunction with the **connect** command helps VSM keep in sync with vCenter Server.
- Each VSM uses a unique extension key to communicate with vCenter Server and perform operations on a DVS.

Extension Key

The VSM uses the extension key when communicating with the vCenter Server. Each VSM has its own unique extension key, such as Cisco_Nexus_1000V_32943215

Use the **show vmware vc extension-key** command to find the extension key of the VSM. It is also listed in the .xml file.

The extension key registered on the vCenter Server can be found through the MOB. For more information, see the "Finding the Extension Key Tied to a Specific DVS" procedure on page 3-8.

The same extension key cannot be used to create more than one DVS on the vCenter Server.

Recovering a DVS

You can use this procedure to recover a DVS if the VSM VM that was used to create it is lost or needs to be replaced. This section includes the following procedures:

- Recovering a DVS With a Saved Copy of the VSM, page 22-3
- Recovering a DVS Without a Saved Copy of the VSM, page 22-4

Recovering a DVS With a Saved Copy of the VSM

You can use this procedure to recover a DVS when you have previously saved a back up copy of the VSM configuration file.

BEFORE YOU BEGIN

Before starting this procedure, you must know or do the following:

- Use this procedure if you have previously saved a back up copy of the VSM configuration file. If you have not previously saved a back up copy, the see the "Recovering a DVS Without a Saved Copy of the VSM" procedure on page 22-4.
- Make sure that the VSM VM switchname is the same as the DVS switchname on the vCenter Server. This allows the VSM configuration to synchronize with the correct DVS on the vCenter Server.

  To change the VSM switchname use the **switchname** *newname* command.

Step 1 From the MOB, find the DVS extension key.
For more information, see the "Finding the Extension Key Tied to a Specific DVS" procedure on page 3-8.

Step 2 On the VSM, add the DVS extension key found in Step 1.

The extension key allows the VSM to log in to the vCenter server.

```
Example:
switch# config t
switch(config)# vmware vc extension-key Cisco_Nexus_1000V_32943215
```

Step 3 From the MOB, unregister the extension key found in Step 1.

For more information, see the "Unregistering the Extension Key in the vCenter Server" procedure on page 3-11.

Step 4 From the VC client, register the extension (plug-in) for the VSM.

For more information see the following procedure in the *Cisco Nexus 1000V Getting Started Guide*.

- Creating a Cisco Nexus 1000V Plug-In on the vCenter Server

Step 5 On the VSM, restore the configuration using a previously saved copy of the VSM configuration file.

copy *path/filename* **running-config**

```
Example:
switch# copy sftp://user1@172.22.36.10/backup/hamilton_cfg running-config
```

Step 6 Do one of the following:

- If the vCenter server connection is not part of the previously saved configuration, continue with the next step.
- Otherwise, go to Step 8.

Step 7 On the VSM, restore the configuration for the vCenter server connection.

```
Example:
switch# config t
switch (config)# svs connection VC
switch(config-svs-conn#) protocol vmware-vim
switch(config-svs-conn#) remote ip address 192.168.0.1
switch(config-svs-conn#) vmware dvs datacenter-name Hamilton-DC
```

Step 8 Connect to vCenter Server.

```
Example:
switch(config-svs-conn#) connect
```

You can now use the old DVS or remove it.

Recovering a DVS Without a Saved Copy of the VSM

You can use this procedure to recover a DVS when you have not previously saved a back up copy of the VSM configuration file.

BEFORE YOU BEGIN

Before starting this procedure, you must know or do the following:

- The folder in which the VSM resides must be:
  - At the root-level of the Data Center in which it resides.
    It cannot be embedded in another folder.
  - Of the same name as the VSM.

  If the folder does not meet the above criteria, the connection to vCenter server fails with the error, *the VSM already exists*.
- Use this procedure if you have not previously saved a back up copy of the VSM configuration file. If you have previously saved a back up copy, then see the "Recovering a DVS With a Saved Copy of the VSM" procedure on page 22-3.
- If you have not previously saved a back up copy of the VSM configuration file, then you may try recreating the old port profiles before connecting to the VC. This procedure has a step for recreating port profiles. If you do not recreate these before connecting to VC, then all the port groups present on the VC are removed and all ports in use are moved to the quarantine port groups.
- Make sure that the VSM VM switchname is the same as the DVS switchname on the vCenter Server. This allows the VSM configuration to synchronize with the correct DVS on the vCenter Server.

  To change the VSM switchname use the **switchname** *newname* command.

Step 1 From the MOB, find the DVS extension key.
For more information, see the "Finding the Extension Key Tied to a Specific DVS" procedure on page 3-8.

Step 2 On the VSM, add the DVS extension key found in Step 1.

The extension key allows the VSM to log in to the vCenter server.

```
Example:
switch# config t
switch(config)# vmware vc extension-key Cisco_Nexus_1000V_32943215
```

Step 3 From the MOB, unregister the extension key found in Step 1.

For more information, see the "Unregistering the Extension Key in the vCenter Server" procedure on page 3-11.

Step 4 From the VC client, register the extension (plug-in) for the VSM.

For more information see the following procedure in the *Cisco Nexus 1000V Getting Started Guide*.

- Creating a Cisco Nexus 1000V Plug-In on the vCenter Server

Step 5 Manually recreate the old port profiles from your previous configuration.

For more information, see the following procedures in the *Cisco Nexus 1000V Getting Started Guide*.

- Configuring the system port profile for VSM-VEM Communication
- Configuring the uplink port profile for VM Traffic
- Configuring the data port profile for VM Traffic

Note If you do not manually recreate the port profiles, then all port groups on the vCenter Server are removed when the VSM connects.

Step 6 On the VSM, restore the configuration for the vCenter server connection.

```
Example:
switch# config t
switch (config)# svs connection VC
switch(config-svs-conn#) protocol vmware-vim
switch(config-svs-conn#) remote ip address 192.168.0.1
switch(config-svs-conn#) vmware dvs datacenter-name Hamilton-DC
```

Step 7 Connect to vCenter Server.

```
Example:
switch(config-svs-conn#) connect
```

You can now use the old DVS or remove it.

Problems Related to VSM and vCenter Server Connectivity

Symptom	Solution
Connections are not supported between Release 4.0(4)SV1(3a) VSMs and VMware vCenter Server 5.0	Upgrade to a compatible version of the Cisco Nexus 1000V software.
The vCenter Server connection seems to succeed, but does not.	Make sure that the domain ID is configured correctly.
The **svs connection command** fails.	Make sure you have configured all parameters for the **svs connection** command.
	Make sure you can ping the vCenter Server IP address.
	Make sure that the proxy.xml file is correct for both the IP address and length.
	Restart the vCenter Server

Symptom	Solution
The connection fails after an ESX reboot	"Connection Failure After ESX Reboot" procedure on page 22-6
The host does not show up in the Add host to DVS screen.	Make sure that the Host is installed with VMware Enterprise plus license containing the Distributed Virtual Switch feature.
Add host to DVS returns an error.	Confirm that the VEM software is installed on the ESX server,
The server name column of the **show module** command output shows the IP address.	The server name shows the host-name or IP address, whichever was used to add the host to the DVS on the vCenter Server.

Example 22-1 shows the **show vms internal event-history errors** command that is useful for examining VC errors in detail. It shows whether an error is caused by a VSM (client) or the server.

Example 22-1 show vms internal event-history error Command

```
switch# show vms internal event-history errors

Event:E_DEBUG, length:239, at 758116 usecs after Tue Feb  3 18:21:58 2009
    [102] convert_soap_fault_to_err(1179): SOAP 1.1 fault: "":ServerFaultCode [VMWARE-VIM]
A DVS switch with spec.name as switch already exists, cannot create DVS switch. A
specified parameter was not correct.spec.name

Event:E_DEBUG, length:142, at 824006 usecs after Tue Feb  3 18:18:30 2009
    [102] convert_soap_fault_to_err(1179): SOAP 1.1 fault: SOAP-ENV:Client [VMWARE-VIM]
Operation could not be completed due to connection failure.

Event:E_DEBUG, length:134, at 468208 usecs after Tue Feb  3 18:15:37 2009
    [102] convert_soap_fault_to_err(1179): SOAP 1.1 fault: "":ServerFaultCode [VMWARE-VIM]
Extension key was not registered before its use.
```

Connection Failure After ESX Reboot

To prevent a loss of connectivity between the VSM and VEM, and preserve a non-default MTU setting for a physical NIC across reboots of the ESX, you must configure a system MTU in the system port profile.

If you use an MTU other than 1500 (the default) for a physical NIC attached to the Cisco Nexus 1000V, then reboots of the ESX can result in a mismatch with the VMware kernel NIC MTU and failure of the VSM and VEM. For example, you may manually configure an MTU of other than 1500 in networks with jumbo frames. During a power cycle, the ESX reboots and the MTU of the physical NIC reverts to the default of 1500 but the VMware kernel NIC does not.

To prevent a loss of connectivity in resulting from an MTU mismatch, see the "Setting the System MTU" procedure on page 22-7.

To recover connectivity if you have not configured system mtu in the system uplink port profile, see

Setting the System MTU

Use this procedure to set a system MTU in your existing system uplink port profiles.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.
- The system port profiles are already configured and you know the uplink profile names.

  For more information, see the *Cisco Nexus 1000V Port Profile Configuration Guide*.
- The MTU size you set for the **system mtu** on the port profile must be less than the size of the **system jumbomtu** configured on the interface.

  For more information about configuring MTU on the interface, see the *Cisco Nexus 1000V Interface Configuration Guide*.
- When you configure a system MTU on a system port profile, it takes precedence over an MTU you may have configured on the interface.
- To verify the ESX MTU settings for corresponding PNICs, use the **ESXcfg-nics -l** command.

SUMMARY STEPS

1. **config t**
2. **port-profile** *profilename*
3. **system mtu** *mtu value*
4. **show port-profil**e [**brief** | **expand-interface** | **usage**] [**name** *profilename*]
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Description
Step 1	**config t** **Example:** switch# config t switch(config)#	Enters global configuration mode.
Step 2	**port-profile** *name* **Example:** switch(config)# port-profile AccessProf switch(config-port-prof)#	Enters port profile configuration mode for the named system uplink port profile.
Step 3	**system mtu** *mtu-size* **Example:** switch(config-port-prof)# system mtu 4000 switch(config-port-prof)#	Designates the MTU size. • Must be an even number between 1500 and 9000. • Must be less than the size of the **system jumbomtu** on the interface.

	Command	Description
Step 4	**show port-profil**e [**brief** | **expand-interface** | **usage**] [**name** *profile-name*] **Example:** switch(config-port-prof)# show port-profile name AccessProf	(Optional) Displays the configuration for verification.
Step 5	**copy running-config startup-config** **Example:** switch(config-port-prof)# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Recovering Lost Connectivity Due to MTU Mismatch

Use this procedure to recover lost connectivity due to an MTU mismatch between the physical NIC and the VMware kernel NIC after an ESX reboot.

BEFORE YOU BEGIN

Before beginning this procedure, you must know or do the following:

- You are logged in to the CLI in EXEC mode.
- To verify the ESX MTU settings for corresponding PNICs, use the **ESXcfg-nics -l** command.

Note Use **vemcmds** only as a recovery measure and then update the MTU value in the port profile configuration for system uplinks or in the interface configuration for non-system uplinks.

SUMMARY STEPS

1. **config t**
2. **module vem** *module_number* **execute vemcmd show port** *port-LTL-number*
3. **module vem** *module_number* **execute vemcmd set mtu** *size* **ltl** *port-LTL-number*

DETAILED STEPS

	Command	Description
Step 1	**config t** **Example:** switch# config t switch(config)#	Enters global configuration mode.
Step 2	**module vem** *module_number* **execute vemcmd show port** *port-LTL-number*	Displays the port configuration including the LTL number needed for Step 3.

	Command	Description

	Example:
	`switch(config)# module vem 3 execute vemcmd show port 48`
	`LTL    IfIndex   Vlan    Bndl  SG_ID Pinned_SGID  Type  Admin State  CBL Mode   Name`
	`17    1a030100    1 T     304      1          32  PHYS  UP    UP       1 Trunk  vmnic1`
	`switch(config)#`	

Step 3	**`module vem`** *`module_number`* **`execute vemcmd set mtu`** *`size`* **`ltl`** *`port-LTL-number`*
	
	Example:
	`switch(config)# module vem 3 execute vemcmd`
	`set mtu 9000 ltl 17`
	`switch(config)#`
		Designates the MTU size for the port, using the LTL number obtained in Step 2.

VSM Creation

Symptom	Possible Causes	Solution
The VSM VM is stuck at the boot prompt.	—	Make sure that you have three e1000 NICs.
The VSM VM cannot ping itself.	—	Configure the management0 interface.
The VSM VM can ping itself, but not the gateway.	—	Make sure the NIC order is correct: control, management, inband/outband.
The VSM VM can ping the gateway, but not the outside subnet.	—	Configure vrf context management.

Port Profiles

When creating a port profile, use the following commands to create the corresponding port groups on the vCenter Server:

- **vmware port-group**
- **state enabled**

Profiles that have the system VLAN configuration allow the VEM to communicate with the VSM.

Make sure that the system port-profile is defined with the right system VLANS.

Use the **show port-profile** and **show port-profile usage** commands to collect basic required information.

Problems with Port Profiles

Symptom	Possible Causes	Solution
You receive an error message "Possible failure in communication with vCenter Server."	The VSM is not connected to the vCenter Server.	Issue the **svs connection vc** command to connect to the vCenter Server.
	The port group name is not unique.	Port group names must be unique within a vCenter Server Datacenter.
Port profile or port groups do not appear on the vCenter Server.	—	Make sure you have issued the **vmware port-group** command and **state enable** command.

Problems with Hosts

Symptom	Solution
You receive an error message, DVS Operation failed for one or more members."	Issue the **vem status -v** command to verify if the VEM is running on the host.
	Issue the **vem unload** command to unload the VEM.
	In the vSphere Client, remove the stale DVS: **1.** Go to the **Host** tab Networking->Configuration-> Distributed Virtual Switch **2.** Click **Remove**.
The host is visible on the vCenter Server, but not the VSM.	Issue the **vemcmd show trunk** command to verify that there is an uplink carrying the control VLAN. The profile applied to the uplink must be a system profile with a control VLAN as a system VLAN.
	Verify the control VLAN in the upstream switch port and the path to the VSM VM. Make sure that one uplink at most carries the control VLAN, or that all uplinks and upstream ports carrying the control VLAN are in port channels.
A module flap occurs.	The VSM may be overloaded. Make sure that you have 1 GB of memory and CPU shares for the VSM VM on the vCenter Server.

Problems with VM Traffic

When troubleshooting problems with intra-host VM traffic, follow these guidelines:

- Make sure that at least one of the VMware virtual NICs is on the correct DVS port group and is connected.
- If the VMware virtual NIC is down, determine if there is a conflict between the MAC address configured in the OS and the MAC address assigned by VMware. You can see the assigned MAC addresses in the vmx file.

When troubleshooting problems with inter-host VM traffic, follow these guidelines:

- Determine if there is exactly one uplink sharing a VLAN with the VMware virtual NIC. If there is more than one, they must be in a port channel.
- Ping a SVI on the upstream switch using the **show intX counters** command.

VEM Troubleshooting Commands

Use the following commands to display VEM information:

- **vemlog** – displays and controls VEM kernel logs
- **vemcmd** – displays configuration and status information
- **vem-support all** – collects support information
- **vem status**– collects status information
- **vem version**– collects version information
- **vemlog show last** *number-of-entries* – displays the circular buffer

Example 22-2 vemlog show last Command

```
[root@ESX-cos1 ~]# vemlog show last 5
Timestamp                  Entry CPU  Mod Lv         Message
Oct 13 13:15:52.615416      1095   1    1  4 Warning vssnet_port_pg_data_ …
Oct 13 13:15:52.620028      1096   1    1  4 Warning vssnet_port_pg_data_ …
Oct 13 13:15:52.630377      1097   1    1  4 Warning svs_switch_state …
Oct 13 13:15:52.633201      1098   1    1  8    Info vssnet new switch …
Oct 13 13:16:24.990236      1099   1    0  0         Suspending log
```

- **vemlog show info** – displays information about entries in the log

Example 22-3 vemcmd show info Command

```
[root@ESX-cos1 ~]# vemlog show info
          Enabled: Yes
    Total Entries: 1092
  Wrapped Entries: 0
     Lost Entries: 0
  Skipped Entries: 0
Available Entries: 6898
 Stop After Entry: Not Specified
```

- **vemcmd help** – displays the type of information you can display

Example 22-4 vemcmd help Command

```
[root@ESX-cos1 ~]# vemcmd help
show card              Show the card's global info
show vlan [vlan]       Show the VLAN/BD table
show bd [bd]           Show the VLAN/BD table
show l2 <bd-number>    Show the L2 table for a given BD/VLAN
show l2 all            Show the L2 table
show port [priv|vsm]   Show the port table
show pc                Show the port channel table
show portmac           Show the port table MAC entries
show trunk [priv|vsm]  Show the trunk ports in the port table
show stats             Show port stats
```

VEM Log Commands

Use the following commands to control the vemlog:

- **vemlog stop** – stops the log
- **vemlog clear** – clear s the log
- **vemlog start** *number-of-entries* – starts the log and stops it after the specified number of entries
- **vemlog stop** *number-of-entries* – stops the log after the next specified number of entries
- **vemlog resume** – starts the log, but does not clear the stop value

Error Messages

On the vSphere Client, you can see error messages under the recent tasks tab. You can find detailed description of the error under the Tasks and Events tab. The same messages are also propagated to the VSM.

Table 22-1 lists error messages that you might see on the VSM.

Table 22-1 Error Messages on the VSM

Error	Description
ERROR: [VMWARE-VIM] Extension key was not registered before its use	This error indicates that VSM extension key is not registered.
ERROR: [VMWARE-VIM] A DVS switch with spec.name as switch already exists, cannot create DVS switch. A specified parameter was not correct. spec.name.	This error is displayed after you enter the first **connect** command, and indicates that a DVS already exists with the same name.
ERROR: [VMWARE-VIM] A DVS switch with spec.extensionKey as Cisco_Nexus_1000V_2055343757 already exists, cannot create DVS new-switch. A specified parameter was not correct. spec.extensionKey	This error is displayed when the VSM tries to create a different DVS after changing the switch name.

Table 22-1 Error Messages on the VSM

Error	Description
ERROR: [VMWARE-VIM] A DVS switch with name as switch already exists, cannot reconfigure DVS test. A specified parameter was not correct. Spec.name	This error indicates that a DVS with the same name already exists.
Warning: Operation succeeded locally but update failed on vCenter server.[VMWARE-VIM] DVPortgroup test port 0 is in use. The resource vim.dvs.DistributedVirtualPort 0 is in use.	This warning is displayed when the VSM tries to delete the port profile if the VSM is not aware of the nics attached to the port groups.

CHAPTER 23

Before Contacting Technical Support

This chapter describes the steps to take before calling for technical support and includes the following sections:

- Cisco Support Communities, page 23-1
- Gathering Information for Technical Support, page 23-1
- Obtaining a File of Core Memory Information, page 23-2
- Copying Files, page 23-3

Note If you purchased Cisco support through a Cisco reseller, contact the reseller directly. If you purchased support directly from Cisco, contact Cisco Technical Support at this URL: http://www.cisco.com/warp/public/687/Directory/DirTAC.shtm

Cisco Support Communities

For additional information, visit one of the following support communities:

- Cisco Support Community for Server Networking
- Cisco Communities: Nexus 1000V

Gathering Information for Technical Support

At some point, you may need to contact your customer support representative or Cisco TAC for some additional assistance. This section outlines the steps that the you should perform prior to contacting your next level of support, so you can reduce the amount of time that you spend resolving the issue.

Note Do not reload the module or the switch at least until you have completed Step 1. Some logs and counters are kept in volatile storage and will not survive a reload.

Step 1 Collect switch information and configuration before and after the issue has been resolved.

Configure your Telnet or SSH application to log the screen output to a text file. Use the **terminal length 0** CLI command and then use the **show tech-support details** CLI command.

Step 2 Capture the exact error codes you see in CLI message logs.

- **show logging log** CLI (displays the error messages)
- **show logging last** *number* (displays the last lines of the log)

Step 3 Answer the following questions before calling for technical support:

- On which switch or port is the problem occurring?
- Which Cisco Nexus 1000V software, driver versions, operating systems versions and storage device firmware are in your fabric?
- ESX and vCenter Server software that you are running?
- What is the network topology?
- Were any changes being made to the environment (VLANs, adding modules, upgrades) prior to or at the time of this event?
- Are there other similarly configured devices that could have this problem, but do not?
- Where was this problematic device connected (which switch and interface)?
- When did this problem first occur?
- When did this problem last occur?
- How often does this problem occur?
- How many devices have this problem?
- Were any traces or debug output captured during the problem time? What troubleshooting steps have you attempted? Which, if any, of the following tools were used?
  - Ethanalyzer, local, or remote SPAN
  - CLI debug commands
  - traceroute, ping

Step 4 Is your problem related to a software upgrade attempt?

- What was the original Cisco Nexus 1000V version?
- What is the new Cisco Nexus 1000V version?

Obtaining a File of Core Memory Information

Cisco customer support engineers often use files from your system for analysis. One such file contains memory information and is referred to as a core dump. The file is sent to a TFTP server or to a flash card in slot0: of the local switch. You should set up your switch to generate this file under the instruction of your customer support representative and send it to a TFTP server so that it can be emailed to them.

To generate a file of core memory information, or a core dump, use the command in the following example.

```
switch# system cores tftp://10.91.51.200/jsmith_cores
switch# show system cores
Cores are transferred to tftp://10.91.51.200/jsmith_cores
```

Note The filename (indicated by jsmith_cores) must exist in the TFTP server directory.

Copying Files

You might be required to move files to or from the switch. These files might include log, configuration, or firmware files.

The Cisco Nexus 1000V always acts as a client. An ftp/scp/tftp session will always originate from the switch and either push files to an external system or pull files from an external system.

```
    File Server: 172.22.36.10
    File to be copied to the switch: /etc/hosts
```

The **copy** CLI command supports four transfer protocols and 12 different sources for files.

```
switch# copy ?
    bootflash: Select source filesystem
    core: Select source filesystem
    debug: Select source filesystem
    ftp: Select source filesystem
    licenses Backup license files
    log: Select source filesystem
    modflash: Select source filesystem
    nvram: Select source filesystem
    running-config Copy running configuration to destination
    scp: Select source filesystem
    sftp: Select source filesystem
    slot0: Select source filesystem
    startup-config Copy startup configuration to destination
    system: Select source filesystem
    tftp: Select source filesystem
    volatile: Select source filesystem
```

Use the following syntax to use secure copy (scp) as the transfer mechanism:

```
    "scp:[//[username@]server][/path]"
```

Copy `/etc/hosts` from 172.22.36.10 using the user `user1`, where the destination would be `hosts.txt`.

```
switch# copy scp://user1@172.22.36.10/etc/hosts bootflash:hosts.txt
user1@172.22.36.10's password:
hosts 100% |*****************************| 2035 00:00
```

Back up the startup configuration to an SFTP server.

```
switch# copy startup-config sftp://user1@172.22.36.10/test/startup-configuration.bak1
Connecting to 172.22.36.10...
User1@172.22.36.10's password:
switch#
```

Tip Back up the startup configuration to a server daily before you make any changes. You can write a short script to be run on the Cisco Nexus 1000V to perform a save and then back up the configuration. The script only needs to contain two commands: **copy running-configuration startup-configuration** and **copy startup-configuration tftp://***server/name*. To execute the script, enter the **run-script** *filename* command.

CHAPTER 24

Network Segmentation Manager

This chapter describes how to identify and resolve problems with Network Segmentation Manger (NSM) and includes the following sections:

Information About Network Segmentation Manager

See the *Cisco Nexus 1000V Network Segmentation Manager Configuration Guide* for more information.

Problems with Network Segmentation Manager

This section includes symptoms, possible causes and solutions for the following problems with Network Segmentation Manager (NSM). The system message for the majority of the problems is logged in vShield Manager or vCloud Director.

Symptom	Possible Causes	Verification and Solution
Registration failure of vShield Manager with Network Segmentation Manager has occurred. A system message is logged in vShield Manager.	vShield Manager is unable to reach NSM.	Verify that the connection between the Cisco Nexus 1000V and VMware vShield Manager is enabled. Check that vShield Manager is able to ping the Cisco Nexus 1000V. If not, reestablish the Layer 2 or Layer 3 connectivity between vShield Manager and the Cisco Nexus 1000V. See the *Cisco Nexus 1000V Network Segmentation Manager Configuration Guide* for more information.
	vShield Manager is unable to authenticate with NSM.	Verify if the username and password are accurate by checking the Virtual Supervisor Module system logs. The following system log will be displayed if the username and password are inaccurate. **`2012 Jan 20 00:49:59 switch %USER-3-SYSTEM_MSG: VALIDATE: user: admin, Authentication failure - validate`** If not, replace the username and password on the in the networking configuration on vShield Manager.
	The NSM feature is not enabled on the Cisco Nexus 1000V.	Verify if the NSM feature is enabled on the Cisco Nexus 1000V. **show feature** If not, enable the NSM feature. **feature network-segmentation-manager**
	HTTPS is not enabled on the Cisco Nexus 1000V.	Check if the browser can connect to https://<vsm-ip>/? If not, enable the HTTPS server on the VSM. **feature http-server**

Symptom	Possible Causes	Verification and Solution
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in the vCloud Director: `Failed to create network segment`	vCloud Director is unable to create the VLAN associated with the network.	1. Verify that the resources are available to create a VLAN by checking the existing number of VLAN. **show vlan summary** If the number of existing VLANs exceeds the number of supported VLANs (2048), then evaluate if there are any of the VLANs that can be removed from the system. 2. Verify that the VLAN pool in vCloud Director does not contain more than 2048 available VLANs.
The network creation triggered from vCloud Directors fail. A system message similar to the following is logged in vCloud Director: `Template could not be inherited on port-profile`	vCloud Director is unable to inherit the port profile associated with the network segment policy onto the port profile created for the network.	1. Verify if the port profile exists. **show running-config port-profile** *name* To identify the name of the port profile, you will need to determine the network segment policy the network was attempting to use. You will need the information about the tenant/organization UUID and the type of network pool the network was being created from (VXLAN or VLAN) to find the corresponding network segment policy that has these values configured. If no network segment policy is configured with these values, then use the default network segment policy to identify the name of the port profile. 2. Check the system logs for a port profile inheritance failure message reported by network segmentation manager. See the *Cisco NX-OS System Messages Reference* for more information.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in the vCloud Director: `Failed to set max-ports`	vCloud Director is unable to set the max ports on the port profile.	Check system logs for a maximum number of port failure message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Network already exists`	A network with the same name already exists in the vCloud Director.	1. Delete the existing network that has the same name. **no port-profile** *network name* 2. Delete the bridge domain with the same name if it exists. **no bridge-domain** *name*

Symptom	Possible Causes	Verification and Solution
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to create port-profile`	The Cisco Nexus 1000V is unable to create the port profile required for the network.	Check system logs for a port profile failure message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.
The network creation triggered from vCloud Directors fails. A system message similar to the following is logged in the vCloud Director: `Template does not exist`	vCloud Director is unable to find the port profile associated with the network segment policy associated with the network.	**1.** Verify if the port profile exists. **show running-config port-profile** *name* To identify the name of the port profile, you will need to determine the network segment policy the network was attempting to use. You need the information about the tenant/organization UUID and the type of network pool the network was being created from (VXLAN or VLAN) to find the corresponding network segment policy that has these values configured. If no network segment policy is configured with these values, use the default network segment policy to identify the name of the port profile. **2.** Check system logs for a port profile failure message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Alias ID not found`	vCloud Director is unable to retrieve the port group ID associated with the port profile created for the network.	Verify that the Virtual Supervisor Module (VSM) has an active SVS connection. **show svs connection** When you enter the command, the output must display **operational status: connected**
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to set port-binding`	vCloud Director is unable to set the port binding on the port profile associated with the network	Check system logs for a port binding failure message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to set vlan`	vCloud Director is unable to set the access VLAN on the port profile associated with the network.	Check system logs for a set VLAN failure message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.

Symptom	Possible Causes	Verification and Solution
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to set vmware port-group`	vCloud Director is unable to set Vmware port group property on the port profile.	Check system logs for a port group property failure message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to set state enabled`	vCloud Director is unable to set the property state on the port profile to enabled.	Check system logs for a state enabled property failure message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to collect svs configuration`	vCloud Director is unable to execute the command. **show svs connection**	Verify that the Virtual Supervisor Module (VSM) has an active SVS connection. **show svs connection** When you enter the command, the output must display “operational status: connected”.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Operational status is missing`	vCloud Director is unable to locate the operational status in the SVS connection.	1. Verify that the Virtual Supervisor Module (VSM) has an active SVS connection. **show svs connection** When you enter the command, the output must display “operational status: connected”. 2. Check system logs for a operational status failure message. See the *Cisco NX-OS System Messages Reference* for more information.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `SVS connection is disconnected`	SVS connection is disconnected.	Verify that the Virtual Supervisor Module (VSM) has an active SVS connection. **show svs connection** When you enter the command, the output must display **operational status: connected**
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to create bridge domain`	vCloud Director is unable to create the bridge dmain associated with the network.	Verify that the feature Segmentation is enabled. **show feature** If not, enable the segmentation feature by using the **feature segmentation** command.

Symptom	Possible Causes	Verification and Solution
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to set segment ID`	vCloud Director is unable to set the segment ID associated with the network.	Verify that the segment ID is not already in use by another bridge domain. **show bridge-domain** Check the error message on the system log to retrieve the segment ID.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to set group IP`	vCloud Director is unable to set the group IP associated with the network.	Verify that the group IP is a valid multicast IP address by checking the system logs for invalid IP address error message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.
The network creation triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to set port-profile description`	vCloud Director is unable to set the description for the port profile associated with the network.	Check system logs for a port profile description failure message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.
The network deletion triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to delete interface using the port-profile`	vCloud Director is unable to delete the interfaces inheriting the port profile.	1. Manually delete the interfaces. 2. In vCenter Server, ensure that the VMs associated with the vApp are powered down. 3. In the VSM enter the **no interface vethernet** *vethernet number* command.
The network deletion triggered from vCloud Director fails. A system message similar to the following is logged in vCloud Director: `Failed to delete the port-profile`	vCloud Director is unable to delete the port profile associated with the network.	1. Manually delete the port profile. 2. Check system logs for a port profile deletion failure message reported by NSM. See the *Cisco NX-OS System Messages Reference* for more information.
An vEthernet interface is administratively down. The interface will be in the NoPortProfile state.	The vEthernet interface is in a quarantine state.	1. Verify the interface is quarantined. **show port-profile sync-status** 2. Bring the interface out of quarantine. **no shutdown** The interface comes back online. 3. Verify if the interface is online. **show interface vethernet**

Network Segmentation Manager Troubleshooting Commands

You can use the commands in this section to troubleshoot problems related to the NSM.

Command	Purpose
show network-segment manager switch	Displays the Cisco Nexus 1000V configured with NSM.
show running-config port-profile	Displays the port profile configuration.
show running-config network-segment policy	Displays the NSM policy configuration.
show network-segment policy usage	Displays the network segmentation policy usage by networks.
show network-segment network	Displays the networks associated with a network segmentation policy.
show network-segment network id *id*	Displays the network IDs associated with a network segmentation policy.
show network-segment network name *name*	Displays the name of the networks associated with a network segmentation policy.
show logging logfile | grep NSMGR	Displays the system logs from the network segmentation manager.

For detailed information about **show** command output, see the *Cisco Nexus 1000V Command Reference*.

CHAPTER 25

VXLANs

This chapter describes how to identify and resolve problems that might occur when implementing Virtual Extensible Local Area Networks (VXLANs) and includes the following sections:

Information About VXLANs

This section includes the following topics:

Overview

A Virtual Extensible LAN creates LAN segments by using an overlay approach with MAC-in-UDP encapsulation and a 24-bit segment identifier in the form of a VXLAN ID. The encapsulation carries the original Layer 2 frame from the virtual machine (VM) that is encapsulated from within the Virtual Ethernet Module (VEM). Each VEM is assigned an IP address that is used as the source IP address when encapsulated MAC frames are sent over the network. You can have multiple VTEPs per VEM that are used as sources for this encapsulated traffic. The encapsulation carries the VXLAN identifier used to scope the MAC address of the payload frame. The VXLAN ID to which a VM belongs is indicated within the port profile configuration of the vNIC and is applied when the VM connects to the network. A VXLAN supports three different modes for broadcast, multicast, and MAC distribution mode transport:

- Multicast Mode—A VXLAN uses an IP multicast network to send broadcast, multicast, and unknown unicast flood frames. When a VM joins a VXLAN segment, the server joins a multicast group. Broadcast traffic from the VM is encapsulated and is sent using the multicast outer destination IP address to all the servers in the same multicast group. Subsequent unicast packets are encapsulated and unicast directly to the destination server without a multicast IP address.
- Unicast-only Mode—A VXLAN uses each VEM's single unicast IP address as the destination IP address to send broadcast, multicast, and unknown unicast flood frames. Broadcast traffic from the VM is replicated to each VEM by encapsulating it with a VXLAN header and the designated IP address as the outer destination IP address.
- MAC Distribution Mode (supported only in unicast mode)—In this mode, the unknown unicast flooding is reduced because the Virtual Supervisor Module (VSM) learns all the MAC addresses from the VEMs in all VXLANs and distributes those MAC addresses with VXLAN Tunnel Endpoint (VTEP) IP mappings to other VEMs.

The VXLAN creates LAN segments by using an overlay approach with MAC in IP encapsulation.

VXLAN Tunnel EndPoint

Each VEM requires at least one IP/MAC pair to terminate VXLAN packets. This IP/MAC address pair is known as the VXLAN Tunnel End Point (VTEP) IP/MAC addresses. The VEM supports IPv4 addressing for this purpose. The IP/MAC address that the VTEP uses is configured when you enter the **capability vxlan** command. You can have a maximum of four VTEPs in a single VEM.

One VTEP per VXLAN segment is designated to receive all broadcast, multicast, and unknown unicast flood traffic for the VEM.

When encapsulated traffic is destined to a VEM that is connected to a different subnet, the VEM does not use the VMware host routing table. Instead, the VTEPs initiate the Address Resolution Protocol (ARP) for remote VEM IP addresses. If the VTEPs in the different VEMs are in different subnets, you must configure the upstream router to respond by using the Proxy ARP.

VXLAN Gateway

VXLAN termination (encapsulation and decapsulation) is supported on virtual switches. As a result, the only endpoints that can connect into VXLANs are VMs that are connected to a virtual switch. Physical servers cannot be in VXLANs and routers or services that have traditional VLAN interfaces cannot be used by VXLAN networks. The only way that VXLANs can currently interconnect with traditional VLANs is through VM-based software routers.

Note Starting with Release 5.2(1)SV3(1.15), Cisco Nexus 1000V for VMware vSphere does not support the VXLAN Gateway feature.

VXLAN Trunks

A VXLAN trunk allows you to trunk multiple VXLANs on a single virtual Ethernet interface. To achieve this configuration, you must encapsulate a VXLAN-VLAN mapping on the virtual Ethernet interface.

VXLAN-VLAN mappings are configured through the Virtual Suervisor Module (VSM) and must always be a 1:1 mapping for each Layer 2 domain. VXLAN-VLAN mappings are applied on a virtual Ethernet interface using a port profile. A single port profile can support multiple VLAN-VXLAN mappings.

VXLAN Border Gateway Protocol Control Plane

The Border Gateway Protocol (BGP) control plane enables the Cisco Nexus 1000V to exchange the VXLAN information collected on the VSM-VTEP flood list across VSMs. The Cisco Nexus 1000V supports BGP peering between 16 VSMs to allow VXLAN segments to reach across servers. BGP runs on the VSM and can exchange VXLAN information with the BGP on any other Cisco Nexus 1000V. The Cisco Nexus 1000V can also be used as a route reflector to exchange a VTEP list between VSMs.

This feature extends the unicast-only mode to a multi-VSM environment using a L2VPN EVPN address family. The VTEP information is not exchanged with the VSMs that are running the old version. They will continue to work in multicast mode (VXLAN 1.0) or unicast-only mode in a single Cisco Nexus 1000V (VXLAN 1.5).

BGP Commands

This example shows how to enable BGP:

```
switch# configure terminal
switch(config)# feature bgp
Cisco Nexus 1000V VXLAN Configuration Guide, Release 5.2(1)SV3(1.1)
24 OL-31596-01
Configuring BGP Control Plane
Configuring BGP
switch(config)# interface control0
switch(config-if)# ip address 14.17.199.1/24
switch(config-if)# vrf context default
switch(config-if)# ip route 0.0.0.0/0 14.17.199.254
switch(config-if)# exit
switch(config)# show feature
Feature Name Instance State
---------------------------- -------- --------
bgp
```

This example shows how to enable BGP with the l2vpn evpn address family:

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router)# router-id 192.169.67.11
switch(config-router)# address-family l2vpn evpn
switch(config-router-af)# copy running-config startup-config
```

This example shows how to configure a BGP peer:

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router)# neighbor 192.0.2.1 remote-as 64497
switch(config-router)# password password1
switch(config-router-neighbor)# description Peer Router B
switch(config-router-neighbor)# address-family l2vpn evpn
switch(config-router-neighbor-af)# send-community extended
switch(config-router-neighbor-af)# copy running-config startup-config
```

This example shows how to configure a BGP peer-session template and apply it to a BGP peer:

```
switch# configure terminal
switch(config)# router bgp 65536
switch(config-router)# template peer-session BaseSession
switch(config-router-stmp)# timers 30 90
switch(config-router-stmp)# exit
switch(config-router)# neighbor 192.168.1.2 remote-as 65536
switch(config-router-neighbor)# inherit peer-session BaseSession
switch(config-router-neighbor)# description Peer Router A
switch(config-router-neighbor)# copy running-config startup-config
```

This example shows how to configure a BGP peer-policy template and apply it to a BGP peer:

```
switch# configure terminal
switch(config)# router bgp 65536
switch(config-router)# template peer-session BasePolicy
switch(config-router-ptmp)# maximum-prefix 20
switch(config-router-ptmp)# exit
switch(config-router)# neighbor 192.168.1.1 remote-as 65536
switch(config-router-neighbor)# address-family l2vpn evpn
switch(config-router-neighbor-af)# inherit peer-policy BasePolicy
switch(config-router-neighbor-af)# copy running-config startup-config
```

This example shows how to configure a BGP peer template and apply it to a BGP peer:

```
switch# configure terminal
switch(config)# router bgp 65536
switch(config-router)# template peer BasePeer
switch(config-router-neighbor)# inherit peer-session BaseSession
switch(config-router-neighbor-af)# inherit peer-policy BasePolicy 1
switch(config-router-neighbor-af)# exita
switch(config-router-neighbor)# exit
switch(config-router)# neighbor 192.168.1.2 remote-as 65536
switch(config-router-neighbor)# inherit peer BasePeer
switch(config-router-neighbor)# copy running-config startup-config
```

This example shows how to display the BGP sessions:

```
vsm# show bgp session
Total peers 1, established peers 1
ASN 65000
VRF default, local ASN 65000
peers 1, established peers 1, local router-id 1.1.1.1
State: I-Idle, A-Active, O-Open, E-Established, C-Closing, S-Shutdown
Neighbor ASN Flaps LastUpDn|LastRead|LastWrit St Port(L/R) Notif(S/R)
14.17.199.2 65000 0 00:04:05|00:00:04|00:00:04 E 61467/179 0/0
```

This example shows how to display the VTEPs that are learned through the BGP:

```
vsm# show bgp l2vpn evpn
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 10, local router ID is 172.23.181.67
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath
Network Next Hop Metric LocPrf Weight Path
Route Distinguisher: 172.23.181.67:5000 (EVI 5000) # RD = <Router-id>:<segment-id>
*>l[3]:[5000]:[4]:[192.168.69.3]/88 #Local VTEP 192.168.69.3
0.0.0.0 100 32768 i
```

```
*>i[3]:[5000]:[4]:[192.168.69.104]/88 #VTEP 192.168.69.104 that are learned from peer
172.23.181.68
172.23.181.68 100 0 i
```

This example shows how to display the detailed output for a specific segment ID or RD:

```
vsm# show bgp l2vpn evpn rd 172.23.181.67:5000
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 10, local router ID is 172.23.181.67
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-best
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath
Network Next Hop Metric LocPrf Weight Path
Route Distinguisher: 172.23.181.67:5000 (EVI 5000)
BGP routing table entry for [3]:[5000]:[4]:[192.168.69.3]/88, version 4
Paths: (1 available, best #1)
Flags: (0x00000a) on xmit-list, is not in l2rib/evpn
Path type: local, path is valid, is best path
AS-Path: NONE, path locally originated
0.0.0.0 (metric 0) from 0.0.0.0 (0.0.0.0)
Origin IGP, MED not set, localpref 100, weight 32768
Extcommunity:
RT:1:5000
Advertised to peers:
172.23.181.68
BGP routing table entry for [3]:[5000]:[4]:[192.168.69.104]/88, version 10
Paths: (1 available, best #1)
Flags: (0x00001a) on xmit-list, is in l2rib/evpn
Path type: internal, path is valid, is best path
Imported from 172.23.181.68:5000:[3]:[5000]:[4]:[192.168.69.104]/88
AS-Path: NONE, path sourced internal to AS
172.23.181.68 (metric 0) from 172.23.181.68 (172.23.181.68)
Origin IGP, MED not set, localpref 100, weight 0
Extcommunity:
RT:1:5000
Not advertised to any peer
```

This example shows how to display the BGP convergence time:

```
switch3# show bgp convergence
Global settings:
BGP start time 00:01:06
Config processing completed 00:00:08 after start
BGP out of wait mode 00:00:30 after start

Information for VRF default
Initial-bestpath timeout: 300 sec, configured 0 sec
First peer up 00:00:29 after start
Bestpath timer not running

   IPv4 Unicast:
   First bestpath signalled 00:00:08 after start
   First bestpath completed 00:00:08 after start

   L2VPN EVPN:
   First bestpath signalled 00:00:30 after start
   First bestpath completed 00:00:30 after start
</>
```

This example shows how to display the VTEP list for a specific VXLAN segment ID or all segments:

```
vsm# show bgp l2vpn evpn evi all VTEP
BGP routing table information for VRF default, address family L2VPN EVPN
BGP table version is 17, local router ID is 192.168.66.10
Status: s-suppressed, x-deleted, S-stale, d-dampened, h-history, *-valid, >-best
```

```
Path type: i-internal, e-external, c-confed, l-local, a-aggregate, r-redist
Origin codes: i - IGP, e - EGP, ? - incomplete, | - multipath
Network Next Hop Metric LocPrf Weight Path
Route Distinguisher: 192.168.66.10:5000 (EVI 5000)
*>i66.100.0.1 192.168.66.100 100 0 i
*>l192.168.69.101 0.0.0.0 100 32768 i
*>i192.168.69.201 192.168.67.10 100 0 i
```

This example shows how to display the VTEP list for a specific VXLAN segment ID or all segments:

```
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 192.168.66.10:5000 (EVI 5000)
BGP routing table entry for [3]:[5000]:[4]:[192.168.69.101]/88, version 2
Paths: (1 available, best #1)
Flags: (0x00000a) on xmit-list, is not in l2rib/evpn
Path type: local, path is valid, is best path
AS-Path: NONE, path locally originated
0.0.0.0 (metric 0) from 0.0.0.0 (0.0.0.0)
Origin IGP, MED not set, localpref 100, weight 32768
Extcommunity:
RT:1:5000
Advertised to peers:
192.168.66.100 192.168.67.10
```

This example shows how to display the bridge domain-to-VTEP mappings that are maintained by the VSM and are pushed to all VEMs:

```
switch# show bridge-domain VTEPs
D: Designated VTEP I:Forwarding Publish Incapable VTEP
Note: (*) Denotes active gateway module
Bridge-domain: vxlan-5000
VTEP Table Version: 13
Port Module VTEP-IP Address VTEP-Flags
-------------------------------------------------------------------------------
Veth5 3 192.168.69.101 (D)
Remote - 66.100.0.1 (DI)
Remote - 192.168.69.201 (DI)
```

This example shows how to display the BGP evpn summary:

```
switch# show bgp l2vpn evpn neighbors 192.168.65.10
BGP summary information for VRF default, address family L2VPN EVPN
BGP router identifier 192.168.67.10, local AS number 1
BGP table version is 6, L2VPN EVPN config peers 2, capable peers 1
3 network entries and 3 paths using 348 bytes of memory
BGP attribute entries [2/264], BGP AS path entries [0/0]
BGP community entries [0/0], BGP clusterlist entries [0/0]
Neighbor V AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd
192.168.65.10 4 1 4011 4013 6 0 0 2d18h 1
```

This example shows how to display the detailed state for a neighbor:

```
switch# show bgp l2vpn evpn neighbors 192.168.65.10
BGP neighbor is 192.168.65.10, remote AS 1, ibgp link, Peer index 1
Inherits peer configuration from peer-template vxlan
BGP version 4, remote router ID 192.168.65.10
BGP state = Established, up for 2d18h
TCP MD5 authentication is enabled
Last read 00:00:24, hold time = 180, keepalive interval is 60 seconds
Last written 00:00:59, keepalive timer expiry due 0.819374
Received 4006 messages, 0 notifications, 0 bytes in queue
Sent 4008 messages, 0 notifications, 0 bytes in queue
Connections established 1, dropped 0
```

```
Last reset by us 2d18h, due to session closed
Last reset by peer never, due to No error

Neighbor capabilities:
Dynamic capability: advertised (mp, refresh, gr) received (mp, refresh, gr)
Dynamic capability (old): advertised received
Route refresh capability (new): advertised received
Route refresh capability (old): advertised received
4-Byte AS capability: advertised received
Address family L2VPN EVPN: advertised received
Graceful Restart capability: advertised received

Graceful Restart Parameters:
Address families advertised to peer:
L2VPN EVPN
Address families received from peer:
L2VPN EVPN
Forwarding state preserved by peer for:
Restart time advertised to peer: 120 seconds
Stale time for routes advertised by peer: 300 seconds
Restart time advertised by peer: 120 seconds

Message statistics:
Sent Rcvd
Opens: 2 1
Notifications: 0 0
Updates: 3 2
Keepalives: 4003 4003
Route Refresh: 0 0
Capability: 0 0
Total: 4008 4006
Total bytes: 76196 76167
Bytes in queue: 0 0

For address family: L2VPN EVPN
BGP table version 6, neighbor version 6
1 accepted paths consume 60 bytes of memory
1 sent paths
Extended community attribute sent to this neighbor
Third-party Nexthop will not be computed.

Local host: 192.168.67.10, Local port: 179
Foreign host: 192.168.65.10, Foreign port: 58283
fd = 39
```

This example shows how to display the detailed state for a VXLAN segment (5000 in this case)

```
switch# show bgp internal evi 5000
BGP L2VPN/EVPN RD Information for 192.168.67.10:5000
VNI ID : 5000 (evi_5000)
#Prefixes Local/BRIB : 1 / 2
BGP EVI Information for evi_5000
EVI ID : 5000 (evi_5000)
RD : 192.168.67.10:5000
Prefixes (local/total) : 1/2
Delete pending : 0
Import pending : 0
Import in progress : 0
Export RTs : 1
Export RT list : 1:5000
Export RT chg/chg-pending : 0/0
Import RTs : 1
Import RT list : 1:5000
Import RT chg/chg-pending : 0/0
```

A few additional commands are as follows:

- **show bgp event-history msgs**
- **show bgp event-history events**

Multi-MAC Capability

You can use multi-MAC addresses to mark a virtual Ethernet interface as capable of sourcing packets from multiple MAC addresses. For example, you can use this feature if you have a virtual Ethernet port and you have enabled VXLAN trunking on it and the VM that is connected to the port bridges packets that are sourced from multiple MAC addresses.

By using this feature, you can easily identify multi-MAC capable ports and handle live migration scenarios correctly for those ports.

Fragmentation

The VXLAN encapsulation overhead is 50 bytes. To prevent performance degradation due to fragmentation, the entire interconnection infrastructure between all VEMs exchange VXLAN packets must be configured to carry 50 bytes more than what the VM VNICs are configured to send. For example, if the default VNIC configuration is 1500 bytes, you must configure the VEM uplink port profile, upstream physical switch port, interswitch links, and any routers to carry a maximum transmission unit (MTU) of at least 1550 bytes. If that is not possible, we recommend that the MTU within the guest VMs you configure to be smaller by 50 bytes.

If you do not configure a smaller MTU, the VEM attempts to notify the VM if it performs Path MTU (PMTU) Discovery. If the VM does not send packets with a smaller MTU, the VM fragments the IP packets. Fragmentation occurs only at the IP layer. If the VM sends a frame that is too large, the frame is to be dropped after VXLAN encapsulation and if the frame does not contain an IP packet.

Scalability

Maximum Number of VXLANs

The Cisco Nexus 1000V supports a total of 4000 and 6144 bridge domains.

```
VSM-DAOX(config-port-prof-srv)# show resource-availability vlan

Maximum number of user VLANs supported: 4093
Number of user VLANs created : 3968
Total number of available user VLANs : 125
Note: Total number of available user VLANs additionally depend on number of
bridge-domains under usage. Please verify the usage of bridge-domains too.

VSM-DAOX(config-port-prof-srv)# show resource-availability bridge-domain
Maximum number of bridge-domains per DVS: 6144
Number of bridge-domains currently created: 5004
Number of bridge-domains available*: 1140
* available bridge-domains do not account for created VLANs
```

Supported Features

This section includes the following topics:

- Jumbo Frames, page 25-9
- Disabling the VXLAN Feature Globally, page 25-9

Jumbo Frames

Jumbo frames are supported by the Cisco Nexus 1000V if there is space on the frame to accommodate the VXLAN encapsulation overhead of at least 50 bytes, and the physical switch/router infrastructure has the capability to transport these jumbo-sized IP packets.

Disabling the VXLAN Feature Globally

As a safety precaution, do not use the no feature segmentation command if there are any ports associated with a VXLAN port profile. You must remove all associations before you can disable this feature. You can use the no feature segmentation command to remove all the VXLAN bridge domain configurations on the Cisco Nexus 1000V.

VXLAN Troubleshooting Commands

Use the following commands to display VXLAN attributes.

This section contains the following topics:

- VSM Commands, page 25-9
- VXLAN Gateway Commands, page 25-11

VSM Commands

You can use the commands in this section to troubleshoot problems related to the VSM.

Command	Purpose
show system internal seg_bd info segment 10000	Displays the ports belonging to a specific segment. See Example 25-1 on page 25-10
show system internal seg_bd info port vethernet 1	Displays the vEthernet bridge domain configuration. See Example 25-2 on page 25-10
show system internal seg_bd info port ifindex 0x1c000050	Displays the vEthernet bridge configuration with ifindex as an argument. See Example 25-3 on page 25-10
show system internal seg_bd info port_count	Displays the total number of bridge domain ports. See Example 25-4 on page 25-10

Command	Purpose
show system internal seg_bd info bd vxlan-home	Displays the bridge domain internal configuration See Example 25-5 on page 25-10
show system internal seg_bd info port	Displays the VXLAN vEthernet information. See Example 25-6 on page 25-10

Example 25-1 show system internal seg_bd info segment 10000

```
switch(config)# show system internal seg_bd info segment 10000
Bridge-domain: A
Port Count: 11
Veth1
Veth2
Veth3
```

Example 25-2 show system internal seg_bd info port vethernet 1

```
switch(config)# show system internal seg_bd info port vethernet 1
Bridge-domain: A
segment_id = 10000
Group IP: 225.1.1.1
```

Example 25-3 show system internal seg_bd info port ifindex 0x1c000050

```
switch(config)# show system internal seg_bd info port ifindex 0x1c000050
Bridge-domain: A
segment_id = 10000
Group IP: 225.1.1.1
```

Example 25-4 show system internal seg_bd info port_count

```
switch(config)# show system internal seg_bd info port_count
Number of ports: 11
```

Example 25-5 show system internal seg_bd info bd vxlan-home

```
switch(config)# show system internal seg_bd info bd vxlan-home

Bridge-domain vxlan-home (2 ports in all)
Segment ID: 5555 (Manual/Active)
Group IP: 235.5.5.5
State: UP               Mac learning: Enabled
is_bd_created: Yes
current state: SEG_BD_FSM_ST_READY
pending_delete: 0
port_count: 2
action: 4
hwbd: 28
pa_count: 0
Veth2, Veth5
switch(config)#
```

Example 25-6 show system internal seg_bd info port

```
switch# show system internal seg_bd info port
if_index = <0x1c000010>
```

```
Bridge-domain vxlan-pepsi
rid = 216172786878513168
swbd = 4098

if_index = <0x1c000040>
Bridge-domain vxlan-pepsi
rid = 216172786878513216
swbd = 4098

switch#
```

VXLAN Gateway Commands

Note Starting with Release 5.2(1)SV3(1.15), Cisco Nexus 1000V for VMware vSphere does not support the VXLAN Gateway feature.

To display VXLAN Gateway information that is attached to the VSM:

```
switch# show module vem
Mod  Ports  Module-Type                       Model               Status
---  -----  --------------------------------  ------------------  ------------
3    7      Virtual Service Module            VXLAN Gateway       ok
```

To display VXLAN Gateway information that is not attached to the VSM:

```
VXLANGW# attach vem
VXLANGW(vem-attach)# ?
  vemcmd     Execute vem command
  vemdpa     Execute vemdpa command
  vemdpalog  Execute vemdpalog command
  vemlog     Execute vemlog command
  vempkt     Execute vempkt command
  vemset     Execute vemset command
switch(vem-attach)#
```

To display VXLAN Gateway mappings:

```
VXGW-switch(vem-attach)# vemcmd show vxlan-gw-mappings
VLAN   Segment   NumProbes     State
---------------------------------------
1821     9001        3         Active
1822     9002        3         Active
Linux(debug)#
Linux(debug)#
Linux(debug)# vemcmd show vxlan
LTL     VSM Port      IP       Seconds since Last   Vem Port
                    Netmask    IGMP Query Received
                    Gateway
(* = IGMP Join Interface/Designated VTEP)
-----------------------------------------------------------
 20        Veth7  17.17.19.111        33        vxlannic0         *
                 255.255.255.0
                    17.17.19.1
```

To display VXLAN Gateway statistics:

```
switch(vem-attach)# vemcmd show vxlan-stats
```

```
  LTL  Ucast   Mcast/Repl   Ucast   Mcast    Total
       Encaps  Encaps       Decaps  Decaps   Drops
   17    8717          173    8334       0     242
switch(vem-attach)#

switch(vem-attach)# vemcmd show vxlan-stats ltl 17
VXLAN Port Stats for LTL 17
Unicast Encapsulations: 8756
Multicast Encapsulations/HeadEnd Replications: 173
Unicast Decapsulations: 8372
Multicast Decapsulations: 0
IP Pre-fragmentations: 0
TSO Processed Packets: 0
ICMP Pkt Too Big msgs from upstream: 0
ICMP Pkt Too Big msgs sent to VM: 0
Packets generated by Head End Replication: 172
```

To display the VXLAN Gateway packet path:

```
switch(vem-attach)# vemlog show all
```

To display the bridge-domain configuration on the VSM:

```
switch# show bridge-domain
Note - This command is common for both gateway and VEM.

Global Configuration:
Mode: Unicast-only
MAC Distribution: Disable
Note - If you have enabled MAC distribution, the above command will display Enable.
Bridge-domain segment-cisco (3 ports in all)
Segment ID: 9001 (Manual/Active)
Mode: Unicast-only (default)
MAC Distribution: Disable (default)
Group IP: NULL
State: UP               Mac learning: Enabled
Veth2, Veth3, Veth5
```

To display the bridge-domain vsteps on the VSM:

```
switch# show bridge-domain VTEPs

D: Designated VTEP     I:Forwarding Publish Incapable VTEP

Note: (*) Denotes active gateway module

Bridge-domain: bd-1776
VTEP Table Version: 40
Port         Module   VTEP-IP Address   VTEP-Flags
------------------------------------------------------------------------------
Veth11       4        172.172.0.134     (D)
Veth66       5        172.172.0.145     (D)
Veth67       5        172.172.1.145

-------------------------------------------------------------------------------
Interface       Module  Serv Inst  Vlan  BD-Name
-------------------------------------------------------------------------------
Ethernet8/1     8       1          1821  vxlan-7001
                8       2          1822  vxlan-7002
                8       3          1823  vxlan-7003
                8       4          1824  vxlan-7004
                8       5          1825  vxlan-7005
switch# sh module VTEPs
```

```
D: Designated VTEP     I:Forwarding Publish Incapable VTEP
A: Active VTEP has duplicates        S: Suppressed duplicate VTEP

Note: (*) Denotes active gateway module

Module   Port         VTEP-IP Address   VTEP-Flags
------------------------------------------------------------------------------
3        Veth2        172.172.1.143     (D)
3        Veth3        172.172.2.143
3        Veth4        172.172.3.143
3        Veth9        172.172.5.143
4        Veth11       172.172.0.134     (D)
Veth68       5        172.172.2.145
Veth69       5        172.172.4.145
```

To display the VLAN-VXLAN mappings programmed on the VSM:

```
switch# show bridge-domain mapping
```

To display the interfaces on the VSM:

```
switch# show module VTEP
```

To display the the bridge domain-to-VTEP mappings that are maintained by the VSM and are pushed to all VEMs:

```
switch# show bridge-domain VTEP
```

To displays the MACs learned on the VSM through VEM distribution:

```
switch# show bridge-domain mac

Bridge-domain: segment-cisco
MAC TABLE Version: 1
Note: You can compare with VEM output using the echo show vxlan version-table command.
MAC Address      Module     Port        VTEP-IP Address  VM-IP Address
------------------------------------------------------------------------------
0050.5683.014e   5          Veth5       10.106.199.117   -
0050.5683.0160   4          Veth2       10.106.199.116   -
0050.5683.0161   4          Veth3       10.106.199.116   -
```

To verify the port configuration on the VSM:

```
switch# show int switchport | begin Vethernet2
Name: Vethernet2
  Switchport: Enabled
  Switchport Monitor: Not enabled
  Operational Mode: access
  Access Mode VLAN: 0 (none)
  Access BD name: segment-cisco
```

To verify the VTEP distribution on the VSM:

```
switch# show bridge-domain segment-cisco VTEPs

D: Designated VTEP     I:Forwarding Publish Incapable VTEP

Bridge-domain: segment-cisco
VTEP Table Version: 2
Note: You can compare the VTEP table version with the echo show vxlan version-table on VEM.
Ifindex     Module  VTEP-IP Address
------------------------------------------------------------------------------
Veth4         4     10.106.199.116(D)
Veth1         5     10.106.199.117(D)
```

```
switch#
```

To verify VXLAN mac-distribution:

```
VSM-DAOX# show bridge-domain mac
Bridge-domain: vxlan6227
MAC Table Count: 1 Next Update Count: 0 Last Update Count: 0
MAC Table Version: 1
MAC Address Module Port VTEP-IP Address
-------------------------------------------------------------------------------
0050.5683.3269 8 Veth1502 192.168.10.6

VSM-DAOX# show bridge-domain vxlan6067 mac
Bridge-domain: vxlan6067
MAC Table Count: 77 Next Update Count: 0 Last Update Count: 0
MAC Table Version: 1
MAC Address Module Port VTEP-IP Address
-------------------------------------------------------------------------------
0050.5683.4c88 6 Veth108 192.168.10.13
0050.5691.01d6 3 Veth177 192.168.10.27
0050.5691.0549 3 Veth695 192.168.10.27
```

Additional **show** commands:

```
show platform fwm errors

show platform fwm info (VTEP | trace | error history)

show platform fwm info error history

show platform fwm event-history msgs

show platform fwm info vlan (all|swbd)
```

VEM Commands

To verify VXLAN vEthernet programming:

```
~ # vemcmd show port segments
                          Native  Seg
  LTL   VSM Port  Mode     SegID   State
   50      Veth5   A       5555   FWD
   51      Veth9   A       8888   FWD
~ #
```

To verify VXLAN VTEP programming:

```
~ # vemcmd show vxlan interfaces
LTL           IP       Seconds since Last
                       IGMP Query Received
(* Interface on which IGMP Joins are sent)
------------------------------------------
 49        10.3.3.3        50          *
 52        10.3.3.6        50
~ #
Use “vemcmd show port vlans” to verify that the VTEPs are in the correct transport VLAN.
```

To verify bridge domain creation on the VEM:

```
~ # vemcmd show bd  bd-name vxlan-home
```

```
BD 31, vdc 1, segment id 5555, segment group IP 235.5.5.5, swbd 4098, 1 ports,
"vxlan-home"
Portlist:
     50  RedHat_VM1.eth0

~ #
```

To verify remote IP learning:

```
~ # vemcmd show l2 bd-name vxlan-home
Bridge domain   31 brtmax 4096, brtcnt 2, timeout 300
Segment ID 5555, swbd 4098, "vxlan-home"
Flags:  P - PVLAN  S - Secure  D - Drop
       Type         MAC Address   LTL   timeout   Flags    PVLAN    Remote IP
    Dynamic   00:50:56:ad:71:4e   305         2                     10.3.3.100
     Static   00:50:56:85:01:5b    50         0                      0.0.0.0

~ #
```

To display statistics:

```
~ # vemcmd show vxlan-stats
  LTL  Ucast   Mcast   Ucast   Mcast    Total
       Encaps  Encaps  Decaps  Decaps   Drops
   49       5   14265       4      15       0
   50       6   14261       4      15     213
   51       1      15       0       0      10
   52       0      11       0       0      15

~ #
```

To display detailed per-port statistics for a VXLAN vEthernet/VTEP:

```
~ # vemcmd show vxlan-stats ltl 51
```

To display detailed per-port-per-bridge domain statistics for a VXLAN VTEP for all bridge domains:

```
~ # vemcmd show vxlan-stats ltl <vxlan_VTEP_ltl> bd-all
```

To display detailed per-port-per-bridge domain statistics for a VXLAN VTEP for a specified bridge domain:

```
~ # vemcmd show vxlan-stats ltl vxlan_VTEP_ltl bd-name bd-name
```

To verify the bridge-domain configuration on the VEM:

```
switch# vemcmd show bd bd-name segment-cisco
Note - Use the module command to check the details of VEM and gateway on the VSM.

BD 26, vdc 1, segment id 9001, segment group IP 0.0.0.0, swbd 4102, 2 ports,
"segment-cisco"
Segment Mode: Unicast
Note: If MAC distribution is enabled, the above command will displays Segment moode as
Unicast MAC distribution
VTEP DSN: 1 , MAC DSN: 1
Note: You can check the VTEP and MAC download sequence numbers using the vemcmd show
vxlan-VTEPs and vemcmd show l2 bd bd-name commands.
Portlist:
     53  RedHat_VM1_112.eth4
     54  RedHat_VM1_112.eth5
~ #
```

To display the MAC address table that shows the MAC addresses delivered by the VSM:

```
switch# vemcmd show l2 bd-name segment-cisco
```

```
Bridge domain   26 brtmax 4096, brtcnt 3, timeout 300
Segment ID 9001, swbd 4102, "segment-cisco"
Flags:  P - PVLAN  S - Secure  D - Drop
       Type         MAC Address    LTL   timeout   Flags    PVLAN    Remote IP    DSN
    SwInsta   00:50:56:83:01:4e   561         0               10.106.199.117   1
     Static   00:50:56:83:01:61    54         0                      0.0.0.0   1
     Static   00:50:56:83:01:60    53         0                      0.0.0.0   1

switch#
```

Displays the port configuration on the VEM:

```
switch# vemcmd show port
  LTL   VSM Port  Admin Link  State  PC-LTL  SGID  Vem Port  Type
   17     Eth4/1      UP    UP    F/B*    561      0    vmnic0
   49             DOWN   UP    BLK       0        RedHat_VM1_112 ethernet7
   50      Veth8   DOWN   UP    BLK       0        RedHat_VM1_112.eth8
   51      Veth4     UP   UP    FWD       0     0      vmk1  VXLAN
   52             DOWN   UP    BLK       0        RedHat_VM1_112.eth6
   53      Veth2     UP   UP    FWD       0        RedHat_VM1_112.eth4
   54      Veth3     UP   UP    FWD       0        RedHat_VM1_112.eth5
  561        Po2     UP   UP    F/B*      0
```

Displays the VTEP distribution on the VEM:

```
switch# vemcmd show vxlan-VTEPs
Bridge-Domain: segment-cisco Segment ID: 9001
Designated Remote VTEP IPs (*=forwarding publish incapable):
10.106.199.117(DSN: 1),
Note: You can compare the download sequence number against the VTEP download sequence
number using the vemcmnd show bd bd-name.
```

Displays if the MAC address table displays the remote IP learning in the segment-cisco bridge domain:

```
switch# vemcmd show l2 bd-name segment-cisco
Note - Use the module command to check the details of VEM and gateway on the VSM.

Bridge domain   26 brtmax 4096, brtcnt 3, timeout 300
Segment ID 9001, swbd 4102, "segment-cisco"
Flags:  P - PVLAN  S - Secure  D - Drop
       Type         MAC Address    LTL   timeout   Flags    PVLAN    Remote IP    DSN
    Dynamic   00:50:56:83:01:4e   561         1               10.106.199.117   0
     Static   00:50:56:83:01:61    54         0                      0.0.0.0   0
     Static   00:50:56:83:01:60    53         0                      0.0.0.0   0
```

To display the VLAN-VXLAN mappings programmed on the VEM:

```
switch# vemcmd show vlan-vxlan mapping
Note - Use the module command to check the details of VEM and gateway on the VSM.
```

To display the multi-MAC capable interfaces on the VEM:

```
Note - Use the module command to check the details of VEM and gateway on the VSM.
switch# vemcmd show multi-mac-capable interfaces
```

VEM Packet Path Debugging

Use the following commands to debug VXLAN traffic from a VM on VEM1 to a VM on VEM2.

- VEM1: Verify that packets are coming into the switch from the segment vEthernet.

  vempkt capture ingress ltl *vxlan_veth*

- VEM1: Verify VXLAN ecapsulation.

  ```
  vemlog debug sflisp all
  vemlog debug sfvnsegment all
  ```

- VEM1: Verify that the remote IP address is learned:

  ```
  vemcmd show l2 bd-name segbdname
  ```

  If the remote IP is not learned, packets are sent multicast encapsulated.

- VEM1: Verify encapsulated packets go out or a uplink.

  Use the **vemcmd show vxlan-encap ltl** *ltl* command to find out which uplink is being used.

  ```
  vempkt capture egress ltl uplink
  ```

- VEM1: Look at statistics for any failures.

  ```
  vemcmd show vxlan-stats all
  vemcmd show vxlan-stats ltl veth/vxlanVTEP
  ```

- VEM2: Verify encapsulated packets are arriving on the uplink.

  ```
  vempkt capture ingress ltl uplink
  ```

- VEM2: Verify VXLAN decapsulation.

  ```
  vemlog debug sflisp all
  vemlog debug sfvnsegment all
  ```

- VEM2: Verify decapsulated packets go out on a VXLAN vEthernet interface.

  ```
  vempkt capture egress ltl vxlan_veth
  ```

- VEM2: Look at statistics for any failures:

  ```
  vemcmd show vxlan-stats all
  vemcmd show vxlan-stats ltl veth/vxlanVTEP
  ```

Use the following commands to debug the VXLAN packet path:

```
switch# module vem 4 execute vemlog debug vssnet all
switch# module vem 4 execute vemlog debug sfsched all
switch# module vem 4 execute vemlog debug sfport all
switch# module vem 4 execute vemlog debug sflisp all
switch# module vem 4 execute vemlog debug sfvnsegment all
```

Use the following commands to debug the VXLAN packet path from the VSM:

```
switch# module vem 4 execute vemdpalog debug if_bridge_rt all
switch# module vem 4 execute vemdpalog debug sfbd all
switch# module vem 4 execute vemdpalog debug sf_dp_threads all
switch# module vem 4 execute vemdpalog debug sfl2agent all
switch# module vem 4 execute vemlog debug sfporttable all
```

You can view the output for all the above logs by using the **module vem 4 execute vemlog show all** command.

VEM Multicast Debugging

Use the following command to debug VEM multicast.

- IGMP state on the VEM:

  ```
  vemcmd show igmp vxlan_transport_vlan detail
  ```

Note This command does not show any output for the segment multicast groups. To save multicast table space, segment groups are not tracked by IGMP snooping on the VEM.

- IGMP queries:

Use the **vemcmd show vxlan interfaces** command to verify that IGMP queries are being received.

- IGMP joins from VTEP:

Use the **vempkt capture ingress ltl** *first_vxlan_VTEP_ltl* command to see if the VMware stack is sending joins.

Use the **vempkt capture egress ltl** *uplink_ltl* command to see if the joins are being sent out to the upstream switch.

VXLAN Data Path Debugging

Use the commands listed in this section to troubleshot VXLAN problems.

This section contains the following topics:

vemlog Debugging

To debug the bridge domain setup or configuration, use the following command:

```
vemlog debug sfbd all
```

To debug the port configuration/CBL/vEthernet LTL pinning, use the following command:

```
vemlog debug sfporttable all
```

(for encapsulated/decapsulated setup and decisions)

```
vemlog debug sfvnsegment all
```

To debug for actual packet editing, VXLAN interface handling, and multicast handling, use the following command:

```
vemlog debug sflisp all
```

To debug multicast joins or leaves on the DPA socket, use the following command:

```
echo "debug dpa_allplatform all" > /tmp/dpafifo
```

To debug the bridge domain configuration, use the following command:

```
echo “debug sfl2agent all” > /tmp/dpafifo
```

To debug the port configuration, use the following command:

echo “debug sfportagent all” > /tmp/dpafifo

To debug hitless reconnect (HR) for capability l2-lisp, use the following command:

echo “debug sfportl2lisp_cache all” > /tmp/dpafifo

To debug CBL programming.

echo “debug sfpixmagent all” > /tmp/dpafifo

To debug a VXLAN agent that interacts with the VSM, use the following command:

echo "debug sfvxlanagent all" > /tmp/dpafifo

To check the VTEP and MAC addresses, use the following command:

```
Jul  1 10:18:20.852679: num_update_timer_interval 3232890
Jul  1 10:18:20.852716: total_report_vsm_count 0
Jul  1 10:18:20.852739: num_act_VTEP_xfaces 1
Jul  1 10:18:20.853108: DPA READY = TRUE
Jul  1 10:18:20.853223: num_swbds 0
Jul  1 10:18:20.853244: missing_swbds 0
Jul  1 10:18:20.853344: get_VTEP_txnid 3
Jul  1 10:18:20.853372: get_mac_txnid 5
Jul  1 10:18:20.853475: vxlan_retry_intvl 600
Jul  1 10:18:20.853602: get_mac_sent = TRUE
Jul  1 10:18:20.853622: get_VTEP_sent = TRUE
Jul  1 10:18:20.853733: VTEP_mismatch_syslog_sent = FALSE
Jul  1 10:18:20.853843: mac_mismatch_syslog_sent = FALSE
Jul  1 10:18:20.853863: delete_notif_rx(Pending MAC deletes) = FALSE
Jul  1 10:18:20.853876: update timer ticks that pending deletes not sent 0
Jul  1 10:18:20.853890: VxLAN update timer state: 1
Jul  1 10:18:20.853906: VSM connected: FALSE
Jul  1 10:18:20.854021: Last retry slot 0 MAC 00:00:00:00:00:00
Jul  1 10:18:20.854132: Last delete slot 0 MAC 00:00:00:00:00:00
Hash      SWBD      VTEP-Ver     MAC-Ver   Created on DP   Need version check
  0       4096         23           0          1           0
  1       4097         23          24          1           0
  2       4098          0           0          0           0
Note: You can compare the MAC version output on the VSM using the show bridge-domain mac
command and VTEP version output on the VSM using the show bridge-domain VTEP command.
```

To check the MAC addresses to be distributed on the VSM, use the following command:

```
Flags:  R - Report to VSM  I - VSM Informed
Add: MAC to be distributed
Delete: MAC to be un-distributed
Stale - Stale entry in VSM
No VTEP - NO VTEP. Entry to be removed from VSM
Wait - Wait for Attach from VSM
BD       MAC Address       if-index         Id      VTEP        Flags  VM IP-cnt   Retry
4097   00:50:56:97:15:dc   0x1c000480  3218000   172.172.0.134   I(Add)        0       1
4097   00:50:56:97:06:89   0x1c0004b0  3218000   172.172.0.134   I(Add)        0       1
```

Vempkt

Vempkt has been enhanced to display the VLAN/SegmentID. Use vempkt to trace the packet path through the VEM.

- Encapsulated: Capture ingress on Seg-VEth LTL – Egress on uplink
- Decapsulated: Capture ingress on uplink – Egress on Seg-VEth LTL

Statistics

To display a summary of per-port statistics, use the following command:

```
vemcmd show vxlan-stats
```

To display detailed per-port statistics for VXLAN VTEP, use the following command:

vemcmd show vxlan-stats ltl *vxlan_VTEP_ltl*

To display detailed per-port statistics for the vEthernet interface in a VXLAN, use the following command:

vemcmd show vxlan-stats ltl *vxlan_veth_ltl*

To display detailed per-port-per-bridge domain statistics for a VXLAN VTEP for all bridge domains, use the following command:

vemcmd show vxlan-stats ltl *vxlan_VTEP_ltl* **bd-all**

To display detailed per-port-per-bridge domain statistics for a VXLAN VTEP for the specified bridge domain, use the following command:

vemcmd show vxlan-stats ltl *vxlan_VTEP_ltl* **bd-name** *bd-name*

To display which VXLAN VTEP is used for encapsulation and subsequent pinning to the uplink port channel for static MAC addresses learned on port, use the following command:

vemcmd show vxlan-encap ltl *vxlan_veth_ltl*

To display which VXLAN VTEP is used for encapsulation and subsequent pinning to the uplink port channel, use the following command:

vemcmd show vxlan-encap mac *vxlan_vm_mac*

show Commands

Command	Result
vemcmd show vxlan interfaces	Displays the VXLAN encapsulated interfaces.
vemcmd show port vlans	Checks the port programming and CBL state for the bridge domain.
vemcmd show bd	Displays the bridge domain segmentId/group/list of ports.
vemcmd show bd bd-name *bd-name-string*	Displays one segment bridge domain.
vemcmd show l2 all	Displays the remote IP being learned.
vemcmd show l2 bd-name *bd-name-string*	Displays the Layer 2 table for one segment bridge domain.
vemcmd show arp all	Displays the IP-MAC mapping for the outer encapsulated header.

CHAPTER 26

VSI Discovery and Configuration Protocol

This chapter describes how to identify and resolve problems that might occur when implementing the VSI Discovery and Configuration Protocol (VDP) and includes the following sections:

- Information About VDP, page 26-1
- Problems with VDP, page 26-2
- VDP Troubleshooting Commands, page 26-2

Information About VDP

VDP on the Cisco Nexus 1000V is an implementation of the IEEE standard 802.1Qbg/D2.2 (Edge Virtual Bridging). VDP can detect and signal the presence of end hosts and exchange capability with an adjacent VDP-capable bridge. VDP serves as a reliable first-hop protocol and communicates the presence of end-host Virtual Machines (VMs) to adjacent leaf nodes on the Cisco Dynamic Fabric Automation (DFA) architecture. In addition to detecting the MAC and IP addresses of the end-host VMs when a host comes up, or during VM mobility events, the VDP triggers auto-configuration of leaf nodes on the DFA architecture to make them ready for more VM traffic.

VDP enables network-based overlays that are a more scalable alternative when compared to the host-based overlays for segmentation and enable access to more than 4000 VLANs in a multi-tenant network. With the VDP configured on the Cisco Nexus 1000V, segmentation support for bridge domains is extended to native encapsulated bridge domains. The original VXLAN-based bridge domains can also coexist with these bridge domains.

For more information about the Cisco DFA architecture, see the *Cisco DFA Solutions Guide*.

Problems with VDP

The following are symptoms, possible causes, and solutions for problems with VDP.

Symptom	Possible Causes	Solution
VDP packets are not received by a leaf switch.	The connected port on the VEM does not have the trunk dynamic port profile.	1. Verify that the connected port on the VEM has the trunk dynamic port profile: **show interface ethernet** *slot/port* 2. If the output of the **show interface ethernet** command does not contain dynamic VLANs, configure the port profile for trunk dynamic mode: a. switch# **configure terminal** b. switch(config)# **port-profile** *name* c. switch(config-port-prof)# **switchport mode trunk** d. switch(config-port-prof)# **switchport trunk dynamic**
VM is associated but it is not pinging.	The encapsulation mode is not native.	Verify that encapsulation mode is native and a valid VLAN value is returned by the leaf switch: **module vem** *module_number* **execute vemcmd show bd** **module vem** *module_number* **execute vemcmd show segment** *segment_id*

VDP Troubleshooting Commands

This section includes the following topics:

- VSM Commands, page 26-2
- VEM Commands, page 26-4

VSM Commands

You can use the commands in this section to troubleshoot problems related to VDP.

Command	Purpose
show evb vsi interface vethernet *interface-number*	Displays if the VDP association sequence is complete for a vEthernet interface. Identify the vEthernet port of the VM and use this command. A VSI state of 3 means that it is associated. See Example 26-1 on page 26-3.
show evb	Displays configured information in the EVB process. See Example 26-2 on page 26-3.

Command	Purpose
show run evb	Displays the running configuration for the EVB segmentation. See Example 26-3 on page 26-3.
show ecp	Displays the configured information for ECP. See Example 26-4 on page 26-3.

EXAMPLES

Example 26-1 show evb vsi interface vethernet Command

```
switch(config)# show evb vsi interface vethernet 40
LTL : 135 [module: 2]
Segment : 30000
MAC : 0050.5693.63A1
IP : 30.0.1.2
VSI State : 3
State Machine State : 7
Rwd Expiry Count : 4621
Last CMD Time : 125
Last RSP Time : 125
```

Example 26-2 show evb Command

```
switch(config)# show evb
Edge Virtual Bridging
Role : VDP Station
VDP Mac Address : 0180.0000.0000
VDP Resource Wait Delay : 22(66 secs)
VDP Reinit Keep Alive : 21(20 secs)
```

Example 26-3 show run evb Command

```
switch(config)# show run evb
evb resource-wait-delay 24
evb reinit-keep-alive 25
ecp retransmission-timer-exponent 15
ecp max-retries 6
```

Example 26-4 show ecp Command

```
switch(config)# show ecp
ECP Max ReTries : 3
ECP Retransmition Timer Exp : 14(163840 micro seconds)
```

VEM Commands

You can use the VEM commands in this section to troubleshoot problems related to VDP.

Command	Purpose
vemcmd show segment *segment-id*	Displays a list of VM interfaces that are a part of a segment and indicates if a segment is configured as VDP (native encapsulation mode). See Example 26-5 on page 26-4.
vemcmd show bd *hwbd*	Displays a list of VM interfaces that are a part of an internal bridge domain and indicates if the bridge domain is configured as VDP (native encapsulation mode). See Example 26-6 on page 26-5.
vemcmd show bd bd-name *bd-name*	Displays a list of VM interfaces that are a part of a configured bridge domain and indicates if the bridge domain is configured as VDP (native encapsulation mode). See Example 26-7 on page 26-5.

EXAMPLES

Example 26-5 vemcmd show segment Command

```
~ # vemcmd show segment 8000
BD 21, vdc 1, segment id 8000, segment group IP 224.9.19.10, encap NATIVE, vff_mode
Anycast,swbd 4098, VLAN 0, 28 ports, "BD-Mcast"
Segment Mode: Multicast
Portlist:
     52  VM-L-13-25-10.eth7
     62  VM-L-13-25-2.eth7
     72  VM-L-13-25-1.eth7
     82  VM-L-13-25-3.eth7
     92  VM-L-13-25-7.eth7
    102  VM-L-13-25-5.eth7
    112  VM-L-13-25-4.eth7
    122  VM-L-13-25-6.eth7
    132  VM-L-13-25-8.eth7
    144  VM-L-14-25-1.eth7

    145  VM-L-14-25-2.eth7
    162  VM-L-14-25-10.eth7
    172  VM-L-14-25-3.eth7
    182  VM-L-13-25-9.eth7
    192  VM-L-14-25-4.eth7
    202  VM-L-14-25-8.eth7
    212  VM-L-14-25-7.eth7
    222  VM-L-14-25-6.eth7
    232  VM-L-14-25-5.eth7
    242  VM-L-14-25-9.eth7

    252  VM-L-15-25-10.eth7
    262  VM-L-15-25-3.eth7
    272  VM-L-15-25-2.eth7
    282  VM-L-15-25-1.eth7
    294  VM-L-15-25-7.eth7
    295  VM-L-15-25-4.eth7
    312  VM-L-15-25-5.eth7
    322  VM-L-15-25-6.eth7
```

Example 26-6 vemcmd show bd Command

```
~ # vemcmd show bd 21
BD 21, vdc 1, segment id 8000, segment group IP 224.9.19.10, encap NATIVE, vff_mode
Anycast,swbd 4098, VLAN 0, 28 ports, "BD-Mcast"
Segment Mode: Multicast
Portlist:
     52  VM-L-13-25-10.eth7
     62  VM-L-13-25-2.eth7
     72  VM-L-13-25-1.eth7
     82  VM-L-13-25-3.eth7
     92  VM-L-13-25-7.eth7
    102  VM-L-13-25-5.eth7
    112  VM-L-13-25-4.eth7
    122  VM-L-13-25-6.eth7
    132  VM-L-13-25-8.eth7
    144  VM-L-14-25-1.eth7

    145  VM-L-14-25-2.eth7
    162  VM-L-14-25-10.eth7
    172  VM-L-14-25-3.eth7
    182  VM-L-13-25-9.eth7
    192  VM-L-14-25-4.eth7
    202  VM-L-14-25-8.eth7
    212  VM-L-14-25-7.eth7
    222  VM-L-14-25-6.eth7
    232  VM-L-14-25-5.eth7
    242  VM-L-14-25-9.eth7

    252  VM-L-15-25-10.eth7
    262  VM-L-15-25-3.eth7
    272  VM-L-15-25-2.eth7
    282  VM-L-15-25-1.eth7
    294  VM-L-15-25-7.eth7
    295  VM-L-15-25-4.eth7
    312  VM-L-15-25-5.eth7
    322  VM-L-15-25-6.eth7
```

Example 26-7 vemcmd show bd bd-name Command

```
~ # vemcmd show bd bd-name BD-Mcast
BD 21, vdc 1, segment id 8000, segment group IP 224.9.19.10, encap NATIVE, vff_mode
Anycast,swbd 4098, VLAN 0, 28 ports, "BD-Mcast"
Segment Mode: Multicast
Portlist:
     52  VM-L-13-25-10.eth7
     62  VM-L-13-25-2.eth7
     72  VM-L-13-25-1.eth7
     82  VM-L-13-25-3.eth7
     92  VM-L-13-25-7.eth7
    102  VM-L-13-25-5.eth7
    112  VM-L-13-25-4.eth7
    122  VM-L-13-25-6.eth7
    132  VM-L-13-25-8.eth7
    144  VM-L-14-25-1.eth7
```

```
145  VM-L-14-25-2.eth7
162  VM-L-14-25-10.eth7
172  VM-L-14-25-3.eth7
182  VM-L-13-25-9.eth7
192  VM-L-14-25-4.eth7
202  VM-L-14-25-8.eth7
212  VM-L-14-25-7.eth7
222  VM-L-14-25-6.eth7
232  VM-L-14-25-5.eth7
242  VM-L-14-25-9.eth7

252  VM-L-15-25-10.eth7
262  VM-L-15-25-3.eth7
272  VM-L-15-25-2.eth7
282  VM-L-15-25-1.eth7
294  VM-L-15-25-7.eth7
295  VM-L-15-25-4.eth7
312  VM-L-15-25-5.eth7
322  VM-L-15-25-6.eth7
```

Cisco TrustSec

This chapter describes how to identify and resolve problems that might occur when configuring Cisco TrustSec and includes the following sections:

- Information About Cisco TrustSec, page 27-1
- Cisco TrustSec Troubleshooting Commands, page 27-1
- Problems with Cisco TrustSec, page 27-5

Information About Cisco TrustSec

The Cisco TrustSec security architecture builds secure networks by establishing clouds of trusted network devices. Each device in the cloud is authenticated by its neighbors. Communication on the links between devices in the cloud is secured with a combination of encryption, message integrity checks, and data-path replay protection mechanisms.

Cisco TrustSec also uses the device and user identification information acquired during authentication for classifying, or coloring, the packets as they enter the network. This packet classification is maintained by tagging packets on ingress to the Cisco TrustSec network so that they can be properly identified for the purpose of applying security and other policy criteria along the data path. The tag, also called the security group tag (SGT), allows the network to enforce the access control policy by enabling the endpoint device to act upon the SGT to filter traffic.

See the *Cisco Nexus 1000V Security Configuration Guide* for more information on the Cisco TrustSec feature on Cisco Nexus 1000V.

Cisco TrustSec Troubleshooting Commands

This section contains the following topics:

- Debugging Commands, page 27-2
- Host Logging Commands, page 27-2
- show Commands, page 27-4

Debugging Commands

Command	Purpose
debug cts authentication	Collects and views logs related to Cisco TrustSec authentication.
debug cts authorization	Collects and views logs related to Cisco TrustSec authorization.
debug cts errors	Collects and views logs related to Cisco TrustSec errors and warning messages.
debug cts messages	Collects and views logs related to Cisco TrustSec messages.
debug cts packets	Collects and views logs related to Cisco TrustSec packets.
debug cts relay	Collects and views logs related to Cisco TrustSec relay functionality.
debug cts sxp	Collects and views logs related to Cisco TrustSec SXP.
debug cts sap	Collects and views logs related to the Cisco TrustSec Security Association Protocol (SAP).
debug cts trace	Collects and views logs related to Cisco TrustSec trace functionality.
show cts internal debug-info	Displays Cisco TrustSec debug information.

Host Logging Commands

You can use the commands in this section to troubleshoot commands related to host logging.

ESX Host Command	Description
echo "logfile enable" > /tmp/dpafifo	Enables DPA debug logging. Logs are output to the **/var/log/vemdpa.log** file.
echo "debug sfctsagent all" > /tmp/dpafifo	Enables TrustSec SXP agent debug logging. Logs are output to the **/var/log/vemdpa.log** file.
vemlog debug sfcts_config all	Enables the data path debug logging and captures logs for the data packets sent between the client and the server.
vemlog debug sfdhcps_config all	Enables the data path debug logging and captures logs for DHCP snooping configuration coming from the VSM. To view the logs, enable DHCP snooping on the Cisco Nexus 1000V.

ESX Host Command	Description
vemlog debug sfdhcps_binding_table all	Enables the data path debug logging and captures logs corresponding to the binding database changes. To view the logs, enable DHCP snooping on the Cisco Nexus 1000V.
vemlog debug sfipdb all	Enables the data path debug logging and captures logs corresponding to the IP database that maintains the IP addresses for all the virtual machines that are being tracked using Cisco TrustSec device tracking. To view the logs, enable Cisco TrustSec device tracking on the Cisco Nexus 1000V.
vemcmd show learnt ip	Displays the Cisco TrustSec configuration on the Cisco Nexus 1000V. See Example 27-1 on page 27-3
vemcmd show cts global	Displays if Cisco TrustSec is enabled on the Cisco Nexus 1000V. See Example 27-2 on page 27-3
vemcmd show cts ipsgt	Displays the Cisco TrustSec configuration on the Cisco Nexus 1000V. See Example 27-3 on page 27-3

Example

Example 27-1 vemcmd show learnt ip Command

```
switch# vemcmd show learnt ip
IP Address LTL VLAN BD
/SegID
10.78.1.76 49 353 7
switch#
```

Example 27-2 vemcmd show cts global Command

```
switch# vemcmd show cts global
CTS Global Configuration:
CTS is: Enabled
CTS Device Tracking is: Enabled
switch#
```

Example 27-3 vemcmd show cts ipsgt Command

```
switch# vemcmd show cts ipsgt
IP Address LTL VLAN BD SGT Learnt
10.78.1.76 49 353 7 6766 Device Tracking
switch#
```

show Commands

See the *Cisco Nexus 1000V Command Reference* for more information on the **show** commands for Cisco TrustSec.

Command	Purpose
show cts	Displays the Cisco TrustSec configuration.
show cts sxp	Displays the SXP configuration for Cisco TrustSec.
show feature	Displays the features available, such as CTS, and whether they are enabled.
show running-configuration cts	Displays the running configuration information for Cisco TrustSec.
show cts device tracking	Displays the Cisco TrustSec device tracking configuration.
show cts ipsgt entries	Display the SXP SGT entries for Cisco TrustSec.
show cts role-based sgt-map	Displays the mapping of the IP address to SGT for Cisco TrustSec.
show cts sxp connection	Displays SXP connections for Cisco TrustSec.
show cts interface delete-hold timer	Displays the interface delete hold timer period for Cisco TrustSec.
show cts internal event-history [error |mem-stats | msgs | sxp]	Displays event logs for Cisco TrustSec.

Problems with Cisco TrustSec

This section includes symptoms, possible causes and solutions for the following problems with Cisco TrustSec.

Symptom	Possible Causes	Verification and Solution
The Cisco Nexus 1000V is unable to form an SXP session with Cisco TrustSec.	There is no connection between the Cisco Nexus 1000V and its peer.	Verify if the Cisco Nexus 1000V is connected to its peer. **ping**
	The Cisco TrustSec SXP is not enabled on the Cisco Nexus 1000V.	Verify if the Cisco TrustSec SXP is enabled on the Cisco Nexus 1000V. **show cts sxp** If not, enable the Cisco TrustSec SXP. **cts sxp enable**
	The password configured on the Cisco Nexus 1000V does not match the password configured on its peer.	Verify if the passwords configured on the Cisco Nexus 1000V matches its peer. **show cts sxp**
	The default source IPv4 address is not configured on the Cisco Nexus 1000V.	Verify if the default source IPv4 address is not configured on the Cisco Nexus 1000V. **show cts sxp**
	The SXP peer is not configured as the listener.	Verify that the SXP peer is configured as the listener. **show cts sxp connection**
Cisco TrustSec SXP is unable to learn any IP-SGT mappings on the Cisco Nexus 1000V.	The Cisco TrustSec device tracking is not enabled on the Cisco Nexus 1000V.	Verify if the Cisco TrustSec device tracking is enabled on the Cisco Nexus 1000V. **show cts device tracking** If not, enable the Cisco TrustSec device tracking. **cts sxp device tracking**
	DHCP snooping is not enabled globally on the Cisco Nexus 1000V.	Verify if DHCP snooping feature is enabled globally on the Cisco Nexus 1000V. **show feature** If not, enable DHCP snooping globally. **feature dhcp** Verify if DHCP snooping is enabled on a VLAN on the Cisco Nexus 1000V. **show ip dhcp snooping** If not, enable DHCP snooping on a VLAN. **ip dhcp snooping vlan** *vlan-list*

CHAPTER 28

vCenter Plug-in

Use this chapter to troubleshoot the vCenter Plug-in functionality.

This chapter includes the following topics:

- Information About vCenter Plug-in, page 28-1
- Prerequisites for VMware vSphere Web Client, page 28-1
- Generating a Log Bundle, page 28-2

Information About vCenter Plug-in

The Cisco Nexus 1000V is a software-based Layer 2 switch for the virtualized server environments that are running VMware ESX. The Cisco Nexus 1000V provides a consistent networking experience across the physical and the virtual environments. It consists of two components: the Virtual Ethernet Module (VEM) that is embedded in the hypervisor and a Virtual Supervisor Module (VSM) that manages the networking policies and the quality of service QoS for the virtual machines.

With earlier releases of the Cisco Nexus 1000V, the system administrators had no visibility into the networking aspects of the Cisco Nexus 1000V. Starting with Cisco NX-OS Release 4.2(1)SV2(1.1), the Cisco Nexus 1000V Plug-in for the VMware vCenter Server (vCenter Plug-in) is supported on the Cisco Nexus 1000V. It provides the server administrators with a holistic view of the virtual network and a visibility into the networking aspects of the Cisco Nexus 1000V.

Starting with Cisco NX-OS Release 4.2(1)SV2(1.1), the vCenter Plug-in is supported on the vSphere Web Clients only. The VMware vSphere Web Client enables you to connect to a VMware vCenter Server system to manage a Cisco Nexus 1000V through a browser. The vCenter Plug-in is installed as a new tab called Cisco Nexus 1000v as part of the user interface in the vSphere Web Client.

With the vCenter Plug-in, the server administrators can export the networking details from the vCenter server, investigate the root cause of and prevent the networking issues, and deploy the virtual machines with the policies. The server administrators can monitor and manage the resources effectively with the network details provided in the vCenter Plug-in.

Prerequisites for VMware vSphere Web Client

Refer to the following prerequisites before configuring the vCenter Plug-in functionality on the Cisco Nexus 1000V:

- VMware vCenter Server 5.0 and/or later release.

- VMware vCenter Web Client 5.1. The vCenter Plug-in does not work with the vSphere 5.0 Web Client.
- The following browsers are supported for version 5.1 of the vSphere Web Client:
  - Microsoft Internet Explorer 7, 8, and 9.
  - Mozilla Firefox 3.6 and later.
  - Google Chrome 14 and later.
- vSphere Web Client requires the Adobe Flash Player version 11.1.0 or later to be installed.
- Make sure that Cisco Nexus 1000V Release 4.2(1)SV2(1.1) is installed and configured to a vCenter.

Generating a Log Bundle

You can collect the diagnostic information for VMware vCenter Server by collecting vSphere log files into a single location.

Step 1 Log in to the Windows server where the VMware vCenter Server is installed.

Step 2 Choose **Start** > **All Programs** > **VMware** > **Generate vSphere Web Client Log Bundle**.

You can use this step to generate the vSphere Web Client log bundles even when you are not able to connect to the vCenter Server using the vSphere Client. The log bundle is generated as a .zip file. See VMware documentation *Collect vSphere Log Files* for more information about collecting the log files.

Note Currently the login to the vCenter Plug-in is available through the administrator account only.

CHAPTER 29

Ethanalyzer

This chapter describes how to use Ethanalyzer as a Cisco NX-OS protocol analyzer tool and includes the following section:

- Using Ethanalyzer, page 29-1

Using Ethanalyzer

Ethanalyzer is a Cisco NX-OS protocol analyzer tool based on the Wireshark (formerly Ethereal) open source code. Ethanalyzer is a command-line version of Wireshark that captures and decodes packets. You can use Ethanalyzer to troubleshoot your network and analyze the control-plane traffic.

Command	Purpose
ethanalyzer local interface *interface*	Captures packets sent or received by the supervisor and provides detailed protocol information. **Note** For all commands in this table, you can use the control, ha-primary, ha-secondary, inband/outband interface (packet interface) or management interface.
ethanalyzer local interface *interface* **limit-captured-frames**	Limits the number of frames to capture.
ethanalyzer local interface *interface* **limit-frame-size**	Limits the length of the frame to capture.
ethanalyzer local interface *interface* **capture-filter**	Filters the types of packets to capture.
ethanalyzer local interface *interface* **display-filter**	Filters the types of captured packets to display.
ethanalyzer local interface *interface* **write**	Saves the captured data to a file.
ethanalyzer local read file	Opens a captured data file and analyzes it.

Ethanalyzer does not capture data traffic that Cisco NX-OS forwards in the hardware. Ethanalyzer uses the same capture filter syntax as tcpdump. For more information, see the following URL:

http://www.tcpdump.org/tcpdump_man.html

For information about the syntax of the display filter, see the following URL:

http://wiki.wireshark.org/DisplayFilters

This example shows captured data (limited to four packets) on the management interface:

```
switch# ethanalyzer local interface mgmt limit-captured-frames 4
Capturing on eth1
2012-10-01 19:15:23.794943 10.78.110.241 -> 72.163.145.51 SSH Encrypted response packet
len=64
2012-10-01 19:15:23.796142 10.78.110.241 -> 72.163.145.51 SSH Encrypted response packet
len=144
2012-10-01 19:15:23.796608 10.78.110.241 -> 72.163.145.51 SSH Encrypted response packet
len=144
2012-10-01 19:15:23.797060 10.78.110.241 -> 72.163.145.51 SSH Encrypted response packet
len=144
4 packets captured
switch#
```

For more information about Wireshark, see the following URL: http://www.wireshark.org/docs/