Cisco Virtual Security Gateway for KVM Troubleshooting Guide, Release 5.2(1)VSG2(1.3)

May 26, 2015

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.

Cisco Virtual Security Gateway for KVM Troubleshooting Guide, Release 5.2(1)VSG2(1.3)

CONTENTS

Overview

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

This chapter includes the following sections:

Overview of the 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 Eliminate each potential problem (from most likely to least likely) until the symptoms disappear.

Overview of Best Practices

Best practices are the recommended steps you should take to ensure the proper operation of your network. We recommend the following general best practices for most networks:

- Maintain a consistent Cisco VSG release across all network devices.
- Refer to the release notes for your Cisco VSG release for the latest features, limitations, and caveats.
- Enable system message logging. See the "Overview of Symptoms" section on page 1-3.

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

Troubleshooting Basics

This section introduces questions to ask when troubleshooting a problem with Cisco VSG 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

By answering the questions in the following sections, you can determine the paths you must 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, too 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 steps:

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 "Overview of Symptoms" 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 "Overview of Symptoms" 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 may use to troubleshoot your specific problem.

Each chapter in this guide may include additional tools and commands 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.

Enter the following commands and examine the outputs:

- **show vsg**
- **show version**

- **show running-config**
- **show logging log**
- **show interfaces brief**
- **show interface data 0**
- **show accounting log**
- **show tech support**
- **show nsc-pa status**
- **show ac-driver statistics**

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 serves users who might have identical problems that are perceived by different indicators. You can 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.

Using a given a set of observable symptoms on a network, you can 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 VSG troubleshooting tools.
- Obtain and analyze protocol traces using Switched Port Analyzer (SPAN) or Ethanalyzer on the command line interface (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 Cisco 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 includes the following topics:

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 vsg %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.*

Each system message has an explanation and recommended action. The action might be as simple as no action required or it might involve a fix or a recommendation to contact technical support as shown in the following example:

Error Message `2009 Apr 29 14:57:23 vsg %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 VSG device to send a copy of the message log to a host for more permanent storage. This feature can be useful if you must examine the logs over a long period of time or when the Cisco VSG device is not accessible.

The example provided in this section shows how to configure a Cisco VSG device to use the syslog facility on a Solaris platform. Although a Solaris host is being used, syslog configuration on all UNIX and Linux systems is very similar.

Syslog uses a facility to determine how the logging 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. The messages could be logged to different files or e-mailed to a particular user. Specifying a severity determines that all messages of that level and greater severity (lower number) will be acted upon by the syslog facility.

Note You should configure the Cisco VSG messages to 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 VSG messages to: /var/adm/nxos_logs

To configure a syslog server, follow these steps:

Step 1 Configure the Cisco VSG syslog policy and server through the Cisco PNSC GUI. See the "Configuring Syslog Policy" section in the *Cisco Prime Network Services Controller GUI Configuration Guide*.

Step 2 Configure the syslog server as follows:

a. Modify /etc/syslog.conf to handle local1 messages. For Solaris, there must be at least one tab 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 a log file.

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

c. Restart the syslog function.

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

d. Verify that the syslog function 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 VSG. This example shows that the system image messages generated are listed on the syslog server. Notice that the IP address of the Cisco VSG 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:
%AAA-6-AAA_ACCOUNTING_MESSAGE: update:171.70.212.30@pts/3:admin:configure terminal ; no
boot system (SUCCESS)
Sep 17 11:07:49 [172.22.36.142.2.2] : 2004 Sep 17 11:17:36 pacific:
%AAA-6-AAA_ACCOUNTING_MESSAGE: update:171.70.212.30@pts/3:admin:Boot Image list set to
bootflash:/nexus-1000v-mzg.VSG1.1.bin
Sep 17 11:07:51 [172.22.36.142.2.2] : 2004 Sep 17 11:17:39 pacific:
%AAA-6-AAA_ACCOUNTING_MESSAGE: update:171.70.212.30@pts/3:admin:configure terminal ; boot
system bootflash:/nexus-1000v-mzg.VSG1.1.bin (SUCCESS)
```

Troubleshooting with Logs

The Cisco VSG generates many types of system messages on the switch and sends them to a syslog server. You can view these messages to determine what events may have led up to the current problem condition that you are facing.

Viewing Logs

You can access and view logs in the Cisco VSG by entering the **show logging ?** command as follows:

```
vsg# 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
```

This example shows how to display the VSG server configuration logs:

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

Troubleshooting Fragmentation/Jumbo Issues

When the Cisco VSG, VEM, and ASA communicate with each other, in a service chain or otherwise, there may be issues related to fragmentation or jumbo frames. You need to make the correct MTU settings to ensure seamless traffic flow and better network performance.

Some of the likely scenarios and maximum transmission unit (MTU) setting recommendations for the Cisco VSG are as follows:

- When the VEM communicates with the Cisco VSG in the Layer 2 mode, an additional header with 62 bytes is added to the original packet. The VEM fragments the packet if it exceeds the uplink MTU. For better performance, increase the MTU of all links between the VEM and the Cisco VSG by 62 bytes to account for packet encapsulation, which occurs for communication between vPath and the Cisco VSG.
- When the VEM communicates with the Cisco VSG in the Layer 3 mode, an additional header with 82 bytes is added to the original packet. Fragmentation is supported in Layer3 mode. On VSM, option is provided to enable and disable L3 fragmentation. By default L3, fragmentation is disabled. If L3 fragmentation is enabled, there is no need to increase the Uplink MTU to accommodate the vPath header.
- If the jumbo frames are enabled in the network, make sure that the MTU of the client and server VMs are reduced by the vPath encapsulation size.
- If the Cisco VSG is deployed on a Virtual Extensible Local Area Network (VXLAN), an additional header with 50 bytes is added to the vPath encapsulation. Adjust the MTU by this value.

The recommended MTU settings for the Cisco ASA 1000V are as follows:

- For fragmentation, use these settings:
  - ASA Inside MTU 9000
  - ASA Outside MTU 9000
  - vPath Path-MTU 1500
- For jumbo frames, use these settings:
  - ASA Inside MTU 9000

  - ASA Outside MTU 9000
  - vPath Path-MTU 8950

Contacting Cisco 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 VSM/VSG and PNSC software
- Version of the ESX and vCenter Server software
- 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 product and support contract from Cisco, contact Cisco for support. Cisco provides Layer 1, Layer 2, and Layer 3 support.

After you have collected this information, see the "Obtaining Documentation and Submitting a Service Request" section on page 1-8.

For more information about the steps to take before calling technical support, see the "Before Contacting Technical Support" section on page 10-1.

Related Documentation

This section contains information about the documentation available for Cisco Virtual Security Gateway and related products.

Cisco Virtual Security Gateway Documentation

The following Cisco Virtual Security Gateway for KVM documents are available on Cisco.com at the following URL:

http://www.cisco.com/en/US/products/ps13095/tsd_products_support_series_home.html

- *Cisco Virtual Security Gateway for KVM Switch Release Notes, Release 5.2(1)VSG2(1.3)*
- *Cisco VSG for KVM, Release 5.2(1)VSG2(1.3) and Cisco PNSC, Release 3.4 Installation Guide*
- *Cisco Virtual Security Gateway for KVM Configuration Guide, Release 5.2(1)VSG2(1.3)*

Cisco Prime Network Services Controller Documentation

The following Cisco Prime Network Services Controller (PNSC) documents are available on Cisco.com at the following URL:

http://www.cisco.com/en/US/products/ps11213/tsd_products_support_series_home.html

Cisco Nexus 1000V Series Switch Documentation

The Cisco Nexus 1000V Series Switch documents are available on Cisco.com at the following URL:

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

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to vsg-docfeedback@cisco.com. We appreciate your feedback.

Obtaining Documentation and Submitting a Service Request

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

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

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

Using Troubleshooting Tools

This chapter describes the troubleshooting tools that are available for the Cisco Virtual Security Gateway (VSG).

This chapter includes the following sections:

Commands

Use the CLI from a local console or remotely use the CLI through a Telnet or Secure Shell (SSH) session. The CLI provides a command structure that is similar to the 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 **show system** command for information about system-level components, including codes, errors, and exceptions. Use the **show system error-id** command to find details on error codes:

```
vsg# 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.

Ping 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. When 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 a destination.

Traceroute

Use traceroute to do the following tasks:

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

The **traceroute** command identifies the path taken on a hop-by-hop basis and includes a time stamp at each hop in both directions. This command tests the connectivity of ports along the path between the generating switch and the switch closest to the destination.

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

Monitoring Processes and CPUs

You can monitor and the CPU status and usage.

This section includes the following topics:

Identifying the Running Processes and Their States

The **show processes** command identifies the running processes and the status of each process as follows:

- 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 dash (-) usually means a daemon that 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. The ER state typically designates a process that has been restarted too many times, which causes the system to classify it as faulty and disable it.

This example shows how to identify the available options for the **show processes** command:

```
vsg# show processes ?
```

```
<CR>
  >       Redirect it to a file
  >>      Redirect it to a file in append mode
  cpu     Show processes CPU Info
  log     Show information about process logs
  memory  Show processes Memory Info
  vdc     Show processes in vdc
  |       Pipe command output to filter
```

This example shows how to display the complete output from the Cisco VSG:

```
vsg# show processes

PID    State  PC        Start_cnt    TTY   Process
-----  -----  --------  -----------  ----  -------------
    1      S  b7f8a468            1     -  init
    2      S         0            1     -  ksoftirqd/0
    3      S         0            1     -  desched/0
    4      S         0            1     -  events/0
    5      S         0            1     -  khelper
   10      S         0            1     -  kthread
   18      S         0            1     -  kblockd/0
   35      S         0            1     -  khubd
  188      S         0            1     -  pdflush
  189      S         0            1     -  pdflush
  190      S         0            1     -  kswapd0
  191      S         0            1     -  aio/0
  776      S         0            1     -  kseriod
  823      S         0            1     -  kide/0
  833      S         0            1     -  ata/0
  837      S         0            1     -  scsi_eh_0
 1175      S         0            1     -  kjournald
 1180      S         0            1     -  kjournald
 1743      S         0            1     -  kjournald
 1750      S         0            1     -  kjournald
 1979      S  b7f6c18e            1     -  portmap
 1992      S         0            1     -  nfsd
 1993      S         0            1     -  nfsd
 1994      S         0            1     -  nfsd
 1995      S         0            1     -  nfsd
 1996      S         0            1     -  nfsd
 1997      S         0            1     -  nfsd
 1998      S         0            1     -  nfsd
 1999      S         0            1     -  nfsd
 2000      S         0            1     -  lockd
 2001      S         0            1     -  rpciod
 2006      S  b7f6e468            1     -  rpc.mountd
 2012      S  b7f6e468            1     -  rpc.statd
 2039      S  b7dd1468            1     -  sysmgr
 2322      S         0            1     -  mping-thread
 2323      S         0            1     -  mping-thread
 2339      S         0            1     -  stun_kthread
 2340      S         0            1     -  stun_arp_mts_kt
 2341      S         0            1     -  stun_packets_re
 2376      S         0            1     -  redun_kthread
 2377      S         0            1     -  redun_timer_kth
 2516      S         0            1     -  sf_rdn_kthread
 2517      S  b7f37468            1     -  xinetd
 2518      S  b7f6e468            1     -  tftpd
 2519      S  b79371b6            1     -  syslogd
 2520      S  b7ecb468            1     -  sdwrapd
 2521      S  b7d6c468            1     -  platform
 2526      S         0            1     -  ls-notify-mts-t
 2539      S  b7eaabe4            1     -  pfm_dummy
```

```
2548      S  b7f836be           1      -  klogd
2555      S  b7c07be4           1      -  vshd
2556      S  b7e4e468           1      -  stun
2557      S  b7af2f43           1      -  smm
2558      S  b7ea0468           1      -  session-mgr
2559      S  b7cb2468           1      -  psshelper
2560      S  b7f75468           1      -  lmgrd
2561      S  b7e69be4           1      -  licmgr
2562      S  b7eb4468           1      -  fs-daemon
2563      S  b7e96468           1      -  feature-mgr
2564      S  b7e44468           1      -  confcheck
2565      S  b7ea8468           1      -  capability
2566      S  b7cb2468           1      -  psshelper_gsvc
2577      S  b7f75468           1      -  cisco
2580      S  b777d40d           1      -  clis
2586      S  b76a340d           1      -  port-profile
2588      S  b7cf9468           1      -  xmlma
2589      S  b7e59497           1      - nsc_pa_intf
2590      S  b7e6c468           1      -  vmm
2591      S  b7b7d468           1      -  vdc_mgr
2592      S  b7e72468           1      -  ttyd
2593      R  b7eda5f5           1      -  sysinfo
2594      S  b7d06468           1      -  sksd
2596      S  b7e82468           1      -  res_mgr
2597      S  b7e48468           1      -  plugin
2598      S  b7bb7f43           1      -  npacl
2599      S  b7e93468           1      -  mvsh
2600      S  b7e01468           1      -  module
2601      S  b78fb40d           1      -  fwm
2602      S  b7e92468           1      -  evms
2603      S  b7e8c468           1      -  evmc
2604      S  b7ec3468           1      -  core-dmon
2605      S  b7e10468           1      -  bootvar
2606      S  b767040d           1      -  ascii-cfg
2607      S  b7ce4be4           1      -  securityd
2608      S  b77bf40d           1      -  cert_enroll
2609      S  b7ce1468           1      -  aaa
2612      S  b7aecf43           1      -  l3vm
2613      S  b7adff43           1      -  u6rib
2614      S  b7addf43           1      -  urib
2615      S  b7dce468           1      -  ExceptionLog
2616      S  b7da8468           1      -  ifmgr
2617      S  b7ea4468           1      -  tcap
2621      S  b75e140d           1      -  snmpd
2637      S  b7f03896           1      -  PMon
2638      S  b7be1468           1      -  aclmgr
2662      S  b7af0f43           1      -  adjmgr
2670      S  b7aecf43           1      -  arp
2671      S  b791c896           1      -  icmpv6
2672      S  b7993f43           1      -  netstack
2746      S  b778d40d           1      -  radius
2747      S  b7f3ebe4           1      -  ip_dummy
2748      S  b7f3ebe4           1      -  ipv6_dummy
2749      S  b789840d           1      -  ntp
2750      S  b7f3ebe4           1      -  pktmgr_dummy
2751      S  b7f3ebe4           1      -  tcpudp_dummy
2755      S  b782740d           1      -  cdp
2756      S  b7b6240d           1      -  dcos-xinetd
2758      S  b7b8d40d           1      -  ntpd
2869      S  b7dd9468           1      -  vsim
2870      S  b797440d           1      -  ufdm
2871      S  b796740d           1      -  sal
2872      S  b793840d           1      -  pltfm_config
2873      S  b782f40d           1      -  monitor
```

```
 2874      S  b7d80468            1      -  ipqosmgr
 2875      S  b7a2827b            1      -  igmp
 2876      S  b7a4340d            1      -  eth-port-sec
 2877      S  b7b29468            1      -  copp
 2878      S  b7ad740d            1      -  eth_port_channel
 2879      S  b7b05468            1      -  vlan_mgr
 2880      S  b767240d            1      -  ethpm
 2921      S  b7d1e468            1      -  msp
 2924      S  b7e8c468            1      -  vsn_service_mgr
 2925      S  b7e25497            1      -  sp
 2926      S  b7832497            1      -  policy_engine
 2927      S  b7e3d497            1      -  inspect
 3064      S  b7f836be            1      1  getty
 3066      S  b7f806be            1     S0  getty
 3091      S  b7f1deee            1      -  pa-httpd.sh
 3092      S  b73da4c7            1      -  svc_sam_vsnAG
 3096      S  b7db7b49            1      -  httpd
 3098      S  b7476be4            1      -  svc_sam_commonA
 3103      S  b70254c7            1      -  svc_sam_dme
 3108      S  b7f1deee            1      -  sam_cores_mon.s
 3150      S  b7db6dcc            1      -  httpd
25835      S  b7b4f40d            1      -  dcos_sshd
25850      S  b78e7eee            1      0  vsh
26766      S  b7f5d468            1      -  sleep
26768      S  b7f5d468            1      -  sleep
26769      R  b7f426be            1      0  more
26770      R  b790ebe4            1      0  vsh
26771      R  b7f716be            1      -  ps
    -     NR         -            0      -  tacacs
    -     NR         -            0      -  dhcp_snoop
    -     NR         -            0      -  installer
    -     NR         -            0      -  private-vlan
    -     NR         -            0      -  scheduler
    -     NR         -            0      -  vbuilder
```

Displaying CPU Usage

You can use the **show processes cpu** command to display CPU usage. The command output includes the following information:

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

This example shows how to display all of the CPU processes:

```
vsg# show processes cpu

PID    Runtime(ms)  Invoked   uSecs  1Sec    Process
-----  -----------  --------  -----  ------  -----------
    1         1519     14917    101    0.0%  init
    2          555     16391     33    0.0%  ksoftirqd/0
    3           96     59084      1    0.0%  desched/0
    4         1469     36858     39    0.0%  events/0
    5           35      2901     12    0.0%  khelper
   10            0        14      3    0.0%  kthread
   18            1       193      9    0.0%  kblockd/0
   35            0         1      3    0.0%  khubd
  188            0         3      0    0.0%  pdflush
```

```
  189           95     13678      6    0.0%  pdflush
  190            0         1      0    0.0%  kswapd0
  191            0         2      1    0.0%  aio/0
  776            0         1      3    0.0%  kseriod
  823            3       138     28    0.0%  kide/0
  833            0         2      2    0.0%  ata/0
  837            0         1      4    0.0%  scsi_eh_0
 1175            0         5     12    0.0%  kjournald
 1180            0         1      5    0.0%  kjournald
 1743            5       194     29    0.0%  kjournald
 1750            0        21     21    0.0%  kjournald
 1979            0        21     25    0.0%  portmap
 1992            0        32     23    0.0%  nfsd
 1993            0        20      4    0.0%  nfsd
 1994            0        20      2    0.0%  nfsd
 1995            0        20      2    0.0%  nfsd
 1996            0        20      1    0.0%  nfsd
 1997            0        20      9    0.0%  nfsd
 1998            0        22      3    0.0%  nfsd
 1999            0        22      3    0.0%  nfsd
 2000            0         2     18    0.0%  lockd
 2001            0         1      1    0.0%  rpciod
 2006            0         1     53    0.0%  rpc.mountd
 2012            1         5    341    0.0%  rpc.statd
 2039          906    148314      6    0.0%  sysmgr
 2322            0         1      9    0.0%  mping-thread
 2323            0         1      3    0.0%  mping-thread
...
```

Displaying CPU and Memory Information

You can use the **show system resources** command to display system-related CPU and memory statistics as follows:

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

This example shows how to display how to display the results of available system resources:

```
vsg# show system resources
Load average:   1 minute: 0.00   5 minutes: 0.00   15 minutes: 0.02
Processes   :   321 total, 1 running
CPU states  :   0.0% user,   0.0% kernel,   100.0% idle
Memory usage:   1944668K total,   1114044K used,    830624K free
                  62340K buffers,  479040K cache
```

Syslog

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

- Logging information for monitoring and troubleshooting.
- Selecting the types of logging information for capture.
- Selecting the destination of the captured logging information.

A syslog can store a chronological log of system messages locally or send the messages 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.

Syslog messages are categorized into seven severity levels from debug to critical events. Severity levels that are reported can be limited 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 and saved to a local file or server.

This section includes the following topics:

- Logging Levels, page 2-7
- Enabling Logging for Telnet or SSH, page 2-7

Logging Levels

The Cisco VSG 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 are 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 Secure Shell (SSH) session.

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

Note When logging to a console session is disabled or enabled, that state is applied to all future console sessions. If you exit and log in again to a new session, the state is preserved. When logging to a Telnet or SSH session that is enabled or disabled, that state applies only to that session. The state is not preserved after you exit the session.

The **no logging console** command is enabled by default. Use this command to disable console logging.

The **terminal monitor** command is disabled by default. Use this command to enable logging for Telnet or SSH.

For more information about configuring syslog, see the *Cisco Virtual Network Management Center GUI Configuration Guide*.

CLI Configuration

This section includes the following topics:

Event Log

This section describes event logs.

This section includes the following topics:

Event Log Configuration Format

The configuration is displayed using this format:

```
[no] event-log inspect {{error | info} | {{ftp {error | info | warn | pkt_trace}} | {rsh
{error | info | pkt_trace}} | {tftp {error | info }}}} [terminal]
```

You can configure event logs for either the inspection process or one of its modules. For example, you can use the **event-log inspect error terminal** command to enable error events for the inspection process and to display these messages on the terminal where the command was entered.

Viewing the Event Log Configuration

You can display the event log configuration by using the **show event-log all** command. This example shows how to display the event logs for all the processes and their modules:

```
vsg# show event-log all
event-log inspect tftp error
event-log inspect rsh error
event-log inspect ftp error terminal
```

```
event-log policy_engine attr-mgr error
event-log service-path sp pkt-error terminal
```

Viewing Event Logs

Event logs are always logged in a process that is specific to the message buffer. Process logging in the event log buffer does not incur any overhead. In addition to using the **show event-log** command, you can display messages on a terminal where the event logs are enabled by using the terminal option, which is useful for reproducing a certain behavior.

The **show** command shows all the processes that are integrated with the event log Cisco VSG infrastructure. You can display inspection event logs using the **show system internal event-log inspect** command. The Cisco VSG event log infrastructure is a layer on top of the Cisco NX-OS event log infrastructure. Event logs can be redirected to a file and exported.

To display event logs on the terminal, use the **terminal** option while configuring the event. Different terminals can view different event logs. For example, use the **event-log inspect ftp info terminal** command to enable the information event logs for the inspection FTP module and to display the logs on the terminal. Use the **event-log inspect rsh error terminal** command to display only the error logs that are related to the RSH module. This command helps to debug various modules at the same time.

Event Log Configuration Persistence

You can save the event log configuration by using the **event-log save config** command. This command allows you to save all of the currently enabled event logs in a file. This file is read at the time of the module/process initialization with the event log infrastructure. The event log configuration that is relevant to the process is reapplied during initialization, which makes the event log configuration persistent across the process/system reboot. Some important things about the event log configuration are as follows:

- Terminal information is not reapplied for process or system restarts because that information might not be applicable.
- The event log configuration is independent of the other Cisco NX-OS configurations. The **copy running-config startup-config** and **show running-config** commands do not save and display the event log configuration.
- The event log configuration is specific to the individual system. In a high-availability setup, the configuration must be set up on both systems.

Configuration and Restrictions

Event logs CLIs for the Cisco VSG are classified based on the process and its modules. This section descrubes event log commands.

This section includes the following topics:

VNS Agent

Virtual Network Service (VNS) agent-related event logs are maintained on the Virtual Supervisor Module (VSM), not on the Cisco VSG.

This section includes the following topics:

Core Module

Core events are those events that are related to port attach, port detach, Internet Protocol Database (IPDB), and to port-profile CLI.

This example shows how to enable/disable error messages for the vns_agent core module:

```
vsm# event-log vns-agent core-error [terminal] ----->enable messages to the terminal
vsm# no event-log vns-agent core-error [terminal] ----->disable messages to the terminal
```

This example shows how to enable/disable informational messages for the vns_agent core module:

```
vsm# event-log vns-agent core-info [terminal] ----->enable messages to the terminal
vsm# no event-log vns-agent core-info [terminal] ----->disable messages to the terminal
```

VPath Module

Because the vPath module works based on core-module events, you should always enable core module event logs before you enable the vPath module events.

This example shows how to enable/disable error messages for the vns_agent vPath module:

```
vsm# event-log vns-agent vpath-error [terminal] ----->enable messages to the terminal
vsm# no event-log vns-agent vpath-error [terminal] ----->disable messages to the terminal
```

This example shows how to enable/disable informational messages for the vns_agent vPath module:

```
vsm# event-log vns-agent vpath-info [terminal] ----->enable messages to the terminal
vsm# no event-log vns-agent vpath-info [terminal] ----->disable messages to the terminal
```

License Module

Because the license module works based on core-module events, you should always enable the core module event logs before enabling the license module.

This example shows how to enable/disable error messages for the vns_agent license module:

```
vsm# event-log vns-agent license-error [terminal] ----->enable messages to the terminal
vsm# no event-log vns-agent license-error [terminal] ----->disable messages to the
terminal
```

This example shows how to enable/disable informational messages for the vns_agent license module:

```
vsm# event-log vns-agent license-info [terminal] ----->enable messages to the terminal
vsm# no event-log vns-agent license-info [terminal] ----->disable messages to the terminal
```

Inspection Process

The inspection process uses event log commands for the inspection process and the File Transfer Protocol (FTP), Remote Shell (RSH), and Trivial File Transfer Protocol (TFTP) modules. These processes are all available on the Cisco VSG.

Use the **event-log inspect error** command to display configuration errors, process initialization errors, and so forth. This example shows how to enable/disable error messages for the inspection process:

```
vsg# event-log inspect error [terminal] ----->enable messages to the terminal
vsg# no event-log inspect error [terminal] ----->disable messages to the terminal
```

This example shows how to enable/disable informational messages for the inspection process:

```
vsg# event-log inspect info [terminal] ----->enable messages to the terminal
vsg# no event-log inspect info [terminal] ----->disable messages to the terminal
```

Use the **event-log inspect ftp error** command to display FTP packet processing errors. This example shows how to enable/disable error messages for the inspection FTP module:

```
vsg# event-log inspect ftp error [terminal] ----->enable messages to the terminal
vsg# no event-log inspect ftp error [terminal] ----->disable messages to the terminal
```

The command output is as follows:

```
Mon Oct  4 15:12:14 2010 ie_ftp: flow (->(ING), 6912), Bad ftp command.

Mon Oct  4 15:12:14 2010 ie_ftp: flow (->(ING), 6912), invalid PORT request / PASV reply.
```

This example shows how to enable/disable informational event log messages for the inspection FTP module:

```
vsg# event-log inspect ftp info [terminal] ----->enable messages to the terminal
vsg# no event-log inspect ftp info [terminal] ----->disable messages to the terminal
```

The command output is as follows:

```
Mon Oct  4 15:12:18 2010 ie_ftp: embryonic connection request (ip, port, proto, pfid, cid,
action, offload) = (192.168.1.20, 40074, tcp, 13569, 6912, 3,1).

Mon Oct  4 15:17:11 2010 ie_ftp: flow (<-(ING), 6912), more reply expected in cmd-reply.
```

This example shows how to enable/disable warning messages for the inspection FTP module:

```
vsg# event-log inspect ftp warn [terminal] ----->enable messages to the terminal
vsg# no event-log inspect ftp warn [terminal] ----->disable messages to the terminal
```

The command output is as follows:

```
Mon Oct  4 15:19:03 2010 ie_ftp: flow (<-(ING), 8192), ftp reply not terminated properly.
```

This example shows how to enable/disable packet trace messages for the inspection FTP module:

```
vsg# event-log inspect ftp pkt_trace [terminal] ----->enable messages to the terminal
vsg# no event-log inspect ftp pkt_trace [terminal] ----->disable messages to the terminal
```

The command output is as follows:

```
Mon Oct  4 15:31:46 2010 ie_ftp: flow (->(ING), 17152), flags(S:)

Mon Oct  4 15:31:54 2010 ie_ftp: flow (->(ING), 17152), cmd (USER)
```

This example shows how to enable/disable error messages for the inspection RSH module:

```
vsg# event-log inspect rsh error [terminal] ----->enable messages to the terminal
vsg# no event-log inspect rsh error [terminal] ----->disable messages to the terminal
```

This example shows how to enable/disable informational messages for the inspection RSH module:

```
vsg# event-log inspect rsh info [terminal] ----->enable messages to the terminal
vsg# no event-log inspect rsh info [terminal] ----->disable messages to the terminal
```

The command output is as follows:

```
Mon Oct  4 15:21:29 2010 ie_rsh: emryonic connection request (ip, port, proto, pfid, cid,
action, offload) = (192.168.1.10, 1021, tcp, 22529, 11264, 3, 1).
```

This example shows how to enable/disable packet trace messages for the inspection RSH module:

```
vsg# event-log inspect rsh pkt_trace [terminal] ----->enable messages to the terminal
vsg# no event-log inspect rsh pkt_trace [terminal] ----->disable messages to the terminal
```

The command output is as follows:

```
Mon Oct  4 15:25:26 2010 ie_rsh: flow (->(ING), 15872), rsh inspect action stop punt
```

This example shows how to enable/disable error messages for the inspection TFTP module:

```
vsg# event-log inspect tftp error [terminal] ----->enable messages to the terminal
vsg# no event-log inspect tftp error [terminal] ----->disable messages to the terminal
```

This example shows how to enable/disable informational messages for the inspection TFTP module:

```
vsg# event-log inspect tftp info [terminal] ----->enable messages to the terminal
vsg# no event-log inspect tftp info [terminal] ----->disable messages to the terminal
```

The command output is as follows:

```
Mon Oct  4 15:27:42 2010 ie_tftp: emryonic connection request (ip, port, proto, pfid, cid,
action, offload) = (192.168.1.10, 32771, udp, 33281, 16640, 3, 1)
```

Service Path Process

The service path processes are available on the Cisco VSG.

This section includes the following topics:

The service path process exposes event log output for the vservice path, flow manager, AC infrastructure modules.

Service Path Module

The **event-log service-path sp error** command can display a failure to initialize the FE, and so forth. This example shows how to enable/disable error messages for the service path module:

```
vsg# event-log service-path sp error [terminal] ----->enable messages to the terminal
vsg# no event-log service-path sp error [terminal] ----->disable messages to the terminal
```

Use the **event-log service-path sp info** command to display FE initialization messages, control path messages, and so forth. This example shows how to enable/disable informational messages for the service path module:

```
vsg# event-log service-path sp info [terminal] ----->enable messages to the terminal
vsg# no event-log service-path sp info [terminal] ----->disable messages to the terminal
```

The **event-log service-path sp pkt-error** command can display failures to read or write a packet, a corrupted packet, and so forth.

This example shows how to enable/disable packet error messages for the service path module:

```
vsg# event-log service-path sp pkt-error [terminal] ----->enable messages to the terminal
vsg# no event-log service-path sp pkt-error [terminal] ----->disable messages to the
terminal
```

The **event-log service-path sp pkt-info** command can display the field description of a packet, where the packet arrived from or going to, decisions taken on the packet, and so forth.

This example shows how to enable/disable packet informational messages for the service path module:

```
vsg# event-log service-path sp pkt-info [terminal] ----->enable messages to the terminal
vsg# no event-log service-path sp pkt-info [terminal] ----->disable messages to the
terminal
```

The **event-log service-path sp pkt-detail** command can display the first few 100 bytes of the incoming packets.

This example shows how to enable/disable detailed packet messages for the service path module:

```
vsg# event-log service-path sp pkt-detail [terminal] ----->enable messages to the terminal
vsg# no event-log service-path sp pkt-detail [terminal] ----->disable messages to the
terminal
```

Service Path Flow Manager

This example shows how to enable/disable the packet messages for the service path flow manager module:

```
vsg# event-log service-path fm error [terminal] ----->enable messages to the terminal
vsg# no event-log service-path fm error [terminal] ----->disable messages to the terminal
```

AC Module

The **event-log service-path ac error** command can display failures to initialize the AC, timer, FD, pending queue, and so forth.

This example shows how to enable/disable error messages for the AC module:

```
vsg# event-log service-path ac error [terminal] ----->enable messages to the terminal
vsg# no event-log service-path ac error [terminal] ----->disable messages to the terminal
```

The **event-log service-path ac info** command can display AC initialization messages, control path messages, and so forth.

This example shows how to enable/disable informational messages for the AC module:

```
event-log service-path ac info [terminal] ----->enable messages to the terminal
no event-log service-path ac info [terminal] ----->disable messages to the terminal
```

Policy Engine Process

The policy engine processes are available on the Cisco VSG.

This section includes the following topic:

Attribute Manager Module

This section describes the attribute manager-related errors.

You can use the **event-log policy-engine attr-mgr error** command to display the policy ID for a PE evaluation lookup based on the VNSP ID, IP address, zone name resolution, attribute fetched, and so forth.

This example shows how to enable/disable error messages for the attribute manager module:

```
vsg# event-log policy-engine attr-mgr error [terminal] ----->enable messages to the
terminal
vsg# no event-log policy-engine attr-mgr error [terminal] ----->disable messages to the
terminal
```

This example shows how to enable/disable informational messages for the attribute manager module:

```
vsg# event-log policy-engine attr-mgr info [terminal] ----->enable messages to the
terminal
vsg# no event-log policy-engine attr-mgr info [terminal] ----->disable messages to the
terminal
```

Restrictions

The event log configuration has the following restrictions:

- Terminal information is not reapplied in case of process restart/system restart bacause it may or may not be applicable.
- Event log configuration is independent of the other Cisco NX-OS configurations. The Cisco NX-OS **copy running-config startup-config** and **show running-config** commands do not save and display event log configuration.
- Event log configuration is specific to the individual system. In the high availability (HA) setup, this configuration must be done on both of the systems.

Show Commands

This section includes the following topics:

VSM Show Commands

This section includes the following topics:

- clear vservice statistics [vlan vlan-num ip ip-addr] [module module-num], page 2-19

show nsc-pa status

You can display the NSC policy agent status by entering the **show nsc-pa status** command.

This example shows how to display the NSC policy agent installation status:

```
vsm# show nsc-pa status
NSC Policy-Agent status is - Installed Successfully. Version 3.2(2b)-vsm
```

show vservice node mac brief

You can display a consolidated view of all vservices in use by using the **show vservice node mac brief** command.

This example shows how to display all vServices in use:

```
vsm# show vservice node mac brief
------------------------------------------------------------------------------
                                  Node Information
------------------------------------------------------------------------------
 ID Type   IP-Address      MAC-Addr          Mode   Fail  State   Module
  2 vsg    10.10.10.202    00:50:56:83:00:1e v-3756 close Alive   66,
  4 vsg    192.168.210.1   b6:5b:3b:37:b2:29 v-3756 close Alive   8,
```

The MAC-ADDR column lists the MAC address of the data0 interface that corresponds to that Cisco VSG (if the VEM can resolve it). If the VEM does not resolve the MAC address, it cannot redirect packets to the Cisco VSG. If a valid MAC address is not shown, check if the Cisco VSG data0 is reachable from the VEM. If there is no valid MAC-ADDR, the possible reasons are as follows:

- The data0 interface on the Cisco VSG is not configured
- The VLAN is not up
- A mismatch has occurred in the VLAN specified in the **vservice** node and the port profile used for the Cisco VSG VM.

FAIL specifies the behavior when the Virtual Ethernet Module (VEM) has no connectivity to the Cisco VSG. The default is Close (packets are dropped). Open means packets are forwarded.

The STATE can be Alive, Unreach, or No Licenses. If Unreach, the MAC-ADDR is not resolved or the module is not up. If multiple VEMs inherit the same VM port profile, those interfaces must pass all checks before the state can be Alive. The MODULE column lists the VEM numbers whose interfaces have inherited this configuration.

show vservice node detail

You can display detailed information for all vServices in use by using the **show vservice node detail** command. Information is displayed for each of the associated VEMs. The command output displays the port profile, security profile, organization, and list of Cisco Nexus 1000V ports that have inherited this configuration. Also displayed are any configuration mismatches between the VSM and VEM missing ports for a given port profile, all ports of a port profile that are not configured with the same security profile, and so forth.

This example shows how to display all vServices in use:

```
VSM-338-STRESS# show vservice node detail
------------------------------------------------------------------------------
                              Node Information
```

```
--------------------------------------------------------------------------------
Node ID:1 Name:ASA
Type:asa IPAddr:33.33.33.34 Fail:close Vlan:756 MTU:NA
Mod State MAC-Addr VVer
6 Alive 00:50:56:88:7c:a7 2

Node ID:2 Name:VPX
Type:adc IPAddr:11.11.11.194 Fail:close L3 MTU:1500
Mod State MAC-Addr VVer
3 Alive -- 2
4 Alive -- 2
5 Alive -- 2
6 Alive -- 2
7 Alive -- 2

Node ID:3 Name:VSG
Type:vsg IPAddr:11.11.11.72 Fail:close L3 MTU:NA
Mod State MAC-Addr VVer
3 Alive -- 2
4 Alive -- 2
5 Alive -- 2

MTU filed is added
```

show vservice port brief

You can display information for each virtual Ethernet (vEth) interface by using the **show vservice port brief** command. By default, all attached vEths are listed. Use the **vethernet** option for output of a specific vEth interface.

This example shows how to display the vEth interfaces:

```
vsm# show vservice port brief
--------------------------------------------------------------------------------
                                  Port Information
--------------------------------------------------------------------------------
PortProfile:access-3770-linux-B-l3
Org:root/Tenant-B
Node:node_200.20.201.183_l3_fclose(200.20.201.183) Profile(Id):sp-web(30)
Veth Mod VM-Name                                 vNIC IP-Address
  31  66 sg-centos-vk-5                            2 172.31.2.105,
```

Ensure that the VM Name value matches the name of the VM associated with this vNIC. For vService Data IP in the brackets in the Node, Profile Name, and Org values, ensure that correct values for this VM are displayed. The Profile ID value should never be zero. For IP addresses, ensure that the list of IP addresses matches the IP addresses configured for the specific vNIC for that VM. If not, use the **vemcmd show learnt** command on all VEMs to display the Internet Protocol Database (IPDB) table.

show vservice connection

You can display vservice connections by using the **show vservice connection** command.

This example shows how to display vservice connections:

```
vsm# show vservice connection
              Actions(Act):
              d - drop                      s - reset
              p - permit                    t - passthrough
              r - redirect                  e - error
              _ - not processed yet         upper case - offloaded
                Flags:
```

```
A - seen ack for syn/fin from src    a - seen ack for syn/fin from dst
E - tcp conn established (SasA done)
F - seen fin from src                f - seen fin from dst
R - seen rst from src                r - seen rst from dst
S - seen syn from src                s - seen syn from dst
T - tcp conn torn down (FafA done)   x - IP-fragment connection

#Path chain-VSGvxlan-Ducativxlan-2
#Module 5
Proto SrcIP[:Port]     SAct DstIP[:Port]    DAct   Flags   Bytes
tcp   172.31.2.107:49389  ?      172.31.2.108:22 rs     S      74

#Path chain-VSGvxlan-Ducativxlan-2a
#Module 4
Proto SrcIP[:Port]    SAct DstIP[:Port]     DAct   Flags    Bytes
tcp 172.31.2.107:49389Pr   172.31.2.108:22    ?   SRr    13
```

show vservice statistics [**vlan** *vlan-num* **ip** *ip-addr*] [**module** *module-num*]

You can display vservice statistics by using the **show vservice statistics** command.

This example shows how to display the vservice statistics:

```
vsm# show vservice statistics
#VSN  VLAN: 756, IP-ADDR: 200.1.1.67
  Module: 3
    #VPath Packet Statistics     Ingress         Egress          Total
    Total Seen                    381295         622662         1003957
    Policy Redirects                   0         120681          120681
    No-Policy Passthru             14830          14835           29665
    Policy-Permits Rcvd                0         120681          120681
    Policy-Denies  Rcvd                0              0               0
    Permit Hits                   366465         487146          853611
    Deny   Hits                        0              0               0
    Decapsulated                       0         120681          120681
    Fail-Open                          0              0               0
    Badport Err                        0              0               0
    vService Config Err                0              0               0
    ARP Resolve Err                    0              0               0
    Encap Err                          0              0               0
    All-Drops                          0              0               0
    Total Rcvd From vService                                    120681
    Non-Cisco Encap Rcvd                                             0
    VNS-Port Drops                                                   0
    Policy-Action Err                                                0
    Decap Err                                                        0
    L2-Frag Sent                                                     0
    L2-Frag Rcvd                                                     0
    L2-Frag Coalesced                                                0

    #VPath Flow Statistics
    Active Flows                       0  Active Connections               0
    Forward Flow Create           120681  Forward Flow Destroy          120681
    Reverse Flow Create           120681  Reverse Flow Destroy          120681
    Flow ID Alloc                 241362  Flow ID Free                  241362
    Connection ID Alloc           120681  Connection ID Free            120681
    L2 Flow Create                     0  L2 Flow Destroy                  0
    L3 Flow Create                     0  L3 Flow Destroy                  0
    L4 TCP Flow Create            241362  L4 TCP Flow Destroy           241362
    L4 UDP Flow Create                 0  L4 UDP Flow Destroy              0
    L4 Oth Flow Create                 0  L4 Oth Flow Destroy              0
    Embryonic Flow Create              0  Embryonic Flow Bloom             0
```

```
  L2 Flow Timeout                  0  L2 Flow Offload                  0
  L3 Flow Timeout                  0  L3 Flow Offload                  0
  L4 TCP Flow Timeout         249934  L4 TCP Flow Offload         120681
  L4 UDP Flow Timeout              0  L4 UDP Flow Offload              0
  L4 Oth Flow Timeout              0  L4 Oth Flow Offload              0
  Flow Lookup Hit             853611  Flow Lookup Miss            241362
  Flow Dual Lookup            998732  L4 TCP Tuple-reuse               0
  Flow Classify Err                0  Flow ID Alloc Err                0
  Conn ID Alloc Err                0  Hash Alloc Err                   0
  Flow Exist                       0  Flow Entry Exhaust               0
  Flow Removal Err                 0  Bad Flow ID Receive              0
  Flow Entry Miss                  0  Flow Full Match Err              0
  Bad Action Receive               0  Invalid Flow Pair                0
  Invalid Connection               0
  Hash Alloc                       0  Hash Free                        0
  InvalFID Lookup                  0  InvalFID Lookup Err              0
  Deferred Delete                  0
Module: 4
  #VPath Packet Statistics     Ingress         Egress          Total
  Total Seen                      9886           9890          19776
  Policy Redirects                   0              0              0
  No-Policy Passthru              9886           9890          19776
  Policy-Permits Rcvd                0              0              0
  Policy-Denies  Rcvd                0              0              0
  Permit Hits                        0              0              0
  Deny   Hits                        0              0              0
  Decapsulated                       0              0              0
  Fail-Open                          0              0              0
  Badport Err                        0              0              0
  vService Config Err                0              0              0
  ARP Resolve Err                    0              0              0
  Encap Err                          0              0              0
  All-Drops                          0              0              0
  Total Rcvd From vService                                         0
  Non-Cisco Encap Rcvd                                             0
  VNS-Port Drops                                                   0
  Policy-Action Err                                                0
  Decap Err                                                        0
  L2-Frag Sent                                                     0
  L2-Frag Rcvd                                                     0
  L2-Frag Coalesced                                                0

  #VPath Flow Statistics
  Active Flows                     0  Active Connections               0
  Forward Flow Create              0  Forward Flow Destroy             0
  Reverse Flow Create              0  Reverse Flow Destroy             0
  Flow ID Alloc                    0  Flow ID Free                     0
  Connection ID Alloc              0  Connection ID Free               0
  L2 Flow Create                   0  L2 Flow Destroy                  0
  L3 Flow Create                   0  L3 Flow Destroy                  0
  L4 TCP Flow Create               0  L4 TCP Flow Destroy              0
  L4 UDP Flow Create               0  L4 UDP Flow Destroy              0
  L4 Oth Flow Create               0  L4 Oth Flow Destroy              0
  Embryonic Flow Create            0  Embryonic Flow Bloom             0
  L2 Flow Timeout                  0  L2 Flow Offload                  0
  L3 Flow Timeout                  0  L3 Flow Offload                  0
  L4 TCP Flow Timeout              0  L4 TCP Flow Offload              0
  L4 UDP Flow Timeout              0  L4 UDP Flow Offload              0
  L4 Oth Flow Timeout              0  L4 Oth Flow Offload              0
  Flow Lookup Hit                  0  Flow Lookup Miss                 0
  Flow Dual Lookup                 0  L4 TCP Tuple-reuse               0
  Flow Classify Err                0  Flow ID Alloc Err                0
  Conn ID Alloc Err                0  Hash Alloc Err                   0
  Flow Exist                       0  Flow Entry Exhaust               0
```

```
    Flow Removal Err                    0  Bad Flow ID Receive                0
    Flow Entry Miss                     0  Flow Full Match Err                0
    Bad Action Receive                  0  Invalid Flow Pair                  0
    Invalid Connection                  0
    Hash Alloc                          0  Hash Free                          0
    InvalFID Lookup                     0  InvalFID Lookup Err                0
    Deferred Delete                     0
```

clear vservice statistics [vlan *vlan-num* ip *ip-addr*] [module *module-num*]

You can clear the vservice statistics by using the **clear vservice statistics** command.

This example shows how to clear vservice statistics:

```
vsm# clear vservice statistics vlan 756 ip 200.1.1.67 module 3
Cleared statistics successfully for specified vservice in module 3
vsm-fcs# show vservice statistics vlan 756 ip 200.1.1.67 module 3
#VSN  VLAN: 756, IP-ADDR: 200.1.1.67
  Module: 3
    #VPath Packet Statistics     Ingress         Egress            Total
    Total Seen                         0              0                0
    Policy Redirects                   0              0                0
    No-Policy Passthru                 0              0                0
    Policy-Permits Rcvd                0              0                0
    Policy-Denies  Rcvd                0              0                0
    Permit Hits                        0              0                0
    Deny   Hits                        0              0                0
    Decapsulated                       0              0                0
    Fail-Open                          0              0                0
    Badport Err                        0              0                0
    vService Config Err                    0              0                0
    ARP Resolve Err                    0              0                0
    Encap Err                          0              0                0
    All-Drops                          0              0                0
    Total Rcvd From vService                                           0
    Non-Cisco Encap Rcvd                                               0
    VNS-Port Drops                                                     0
    Policy-Action Err                                                  0
    Decap Err                                                          0
    L2-Frag Sent                                                       0
    L2-Frag Rcvd                                                       0
    L2-Frag Coalesced                                                  0

    #VPath Flow Statistics
    Active Flows                        0  Active Connections                 0
    Forward Flow Create                 0  Forward Flow Destroy               0
    Reverse Flow Create                 0  Reverse Flow Destroy               0
    Flow ID Alloc                       0  Flow ID Free                       0
    Connection ID Alloc                 0  Connection ID Free                 0
    L2 Flow Create                      0  L2 Flow Destroy                    0
    L3 Flow Create                      0  L3 Flow Destroy                    0
    L4 TCP Flow Create                  0  L4 TCP Flow Destroy                0
    L4 UDP Flow Create                  0  L4 UDP Flow Destroy                0
    L4 Oth Flow Create                  0  L4 Oth Flow Destroy                0
    Embryonic Flow Create               0  Embryonic Flow Bloom               0
    L2 Flow Timeout                     0  L2 Flow Offload                    0
    L3 Flow Timeout                     0  L3 Flow Offload                    0
    L4 TCP Flow Timeout                 0  L4 TCP Flow Offload                0
    L4 UDP Flow Timeout                 0  L4 UDP Flow Offload                0
    L4 Oth Flow Timeout                 0  L4 Oth Flow Offload                0
    Flow Lookup Hit                     0  Flow Lookup Miss                   0
    Flow Dual Lookup                    0  L4 TCP Tuple-reuse                 0
    Flow Classify Err                   0  Flow ID Alloc Err                  0
```

```
Conn ID Alloc Err                0  Hash Alloc Err                   0
Flow Exist                       0  Flow Entry Exhaust               0
Flow Removal Err                 0  Bad Flow ID Receive              0
Flow Entry Miss                  0  Flow Full Match Err              0
Bad Action Receive               0  Invalid Flow Pair                0
Invalid Connection               0
Hash Alloc                       0  Hash Free                        0
InvalFID Lookup                  0  InvalFID Lookup Err              0
Deferred Delete                  0
```

Cisco VSG show Commands

The attribute manager maintains a set of tables and does a lookup that is based on the fields in the packet. There are three main tables: DV port table, VM table, and VNSP table. Use the **show vsg dvport** command to display runtime information for the DV port table. For the other two tables, use the **show vsg vm** and **show vsg vnsp** commands.

Hash tables are maintained based on IP addresses (IP address to DV port entry) and VNSP ID (VNSP ID to VNSP entry). An IP address is used when fetching attributes (custom and VM attributes) that are based on the source or destination IP address. It is also used to determine which policy set to evaluate for a given traffic type. The VNSP ID is used (valid VNSP ID in the packet header) to determine which policy set to evaluate. Custom attributes can also be fetched.

This section includes the following topics:

show nsc-pa status

Enter the **show nsc-pa status** command to display the Cisco PNSC policy agent status.

This example shows how to display the Cisco PNSC policy agent status:

```
vsg# show nsc-pa status
NSC Policy-Agent status is - Installed Successfully. Version 2.1(2a)-vsg
```

show service-path statistics

You can display the following statistics that pertain to one vPath by using the **show service-path statistics** command:

- The packets seen by the service path from the vPath.
- Flows created by the service path due to these packets.
- Packets dropped in the service path due to various errors.

Note If no module is given, the command displays the aggregate statistics of all the modules in the given SVS domain.

This command provides the following keyword filters:

- **svs-domain-id** *domain-id*—Displays only the Cisco VSG connections that are associated to the svs-domain specified in the *domain-id*.
- **module** *module-num*—Displays only the Cisco VSG connections that are associated to the svs-domain and VEM module specified in the *domain-id* and the *module-num*. Use this keyword filter only with the **svs-domain-id** filter.

This example shows how to display the statistics using the **svs-domain-id** keyword filter:

```
vsg# show service-path statistics svs-domain-id 118 module 5
Input Packet               161359233  Output Packet              161359220
Vpath Ingress Packet         7608059  Vpath Egress Packet        153751174
Vpath Frag                         0  vService Offload Packet            0
ARP Packet                         0  Unknown L2 Packet                  0
802.3 Packet                       0  Vpath Jumbo Frame                  0
IPV4 Packet                161359233  IPV4 options Packet                0
IPV4 Frag                          0  Unknown L3Proto Packet             0
ICMP Packet                       66  IGMP Packet                        0
TCP Packet                 161359095  UDP Packet                        72
Policy Lookup Packet       160669149  Inspect FTP Packet                 0
Inspect RSH Packet                 0  Inspect TFTP Packet                0
Policy Lookup Fail                 0  Policy Lookup Drop                 0
Inspect FTP Fail                   0  Inspect FTP Drop                   0
Inspect RSH Fail                   0  Inspect RSH Drop                   0
Inspect TFTP Fail                  0  Inspect TFTP Drop                  0
Malformed Packet                   0  Output Fail                        0
Active Flows                  473278  Active Connections            379521
Forward Flow Create          8690219  Forward Flow Destroy         3008524
Reverse Flow Create          3362016  Reverse Flow Destroy         8570433
Flow ID Alloc               12052235  Flow ID Free                11578957
Connection ID Alloc          3362016  Connection ID Free           2982495
L2 Flow Create                     0  L2 Flow Destroy                    0
L3 Flow Create                    66  L3 Flow Destroy                   66
L4 TCP Flow Create          12052097  L4 TCP Flow Destroy         11578819
L4 UDP Flow Create                72  L4 UDP Flow Destroy               72
L4 Other Flow Create               0  L4 Other Flow Destroy              0
Embryonic Flow Create              0  Embryonic Flow Bloom               0
L2 Flow Timeout                    0  L2 Flow Offload                    0
L3 Flow Timeout                   99  L3 Flow Offload                   66
L4 TCP Flow Timeout         25158984  L4 TCP Flow Offload         160668998
L4 UDP Flow Timeout               108  L4 UDP Flow Offload               72
L4 Other Flow Timeout              0  L4 Other Flow Offload              0
```

```
Flow Lookup Hit            157997217  Flow Lookup Miss            12052235
Flow Dual Lookup           138932556  L4 TCP Tuple-reuse         151978861
Flow Classify Err                 0  Flow ID Alloc Err                  0
Conn ID Alloc Err                 0  Hash Alloc Err                     0
Flow Exist                        0  Flow Entry Exhaust                 0
Flow Removal Err                  0  Bad Flow ID receive                0
Flow Entry Missing                0  Flow Full Match Err                0
Bad Action Received               0  Invalid Flow Pair                  0
Invalid Connection                0
```

clear service-path statistics

You can clear the service path statistics globally by using the **clear service-path statistics** command when no option is given. When the SVS domain ID and the module are provided, entering the command clears the statistics of the specified module.

This command provides the following keyword filters:

- **svs-domain-id** *domain-id*—Displays only the Cisco VSG connections that are associated to the svs-domain specified in the *domain-id*.
- **module** *module-num*—Displays only the Cisco VSG connections that are associated to the svs-domain and VEM module specified in the *domain-id* and the *module-num*. Use this only with the **svs-domain-id** filter.

This example shows how to clear the service path statistics:

```
vsg# clear service-path statistics
```

show service-path connection

You can display the connections (flow-table) maintained in the Cisco VSG by using the **show service-path connection** command. These connections are provided per VEM module per SVS domain.

This command provides the following keyword filters:

- **svs-domain-id** *domain-id*—Displays only the Cisco VSG connections that are associated to the svs-domain specified in the *domain-id*.
- **module** *module-num*—Displays only the Cisco VSG connections that are associated to the svs-domain and VEM module specified in the *domain-id* and the *module-num*. Use this keyword filter only with the **svs-domain-id** keyword filter.

This example shows how to display the connections in the Cisco VSG:

```
vsg# show service-path connection
Flags:
P - policy at src                      p - policy at dst
O - conn offloaded to ser-path at src  o - conn offloaded to ser-path at dst
S - seen syn from src                  s - seen syn from dst
A - seen ack for syn/fin from src      a - seen ack for syn/fin from dst
F - seen fin from src                  f - seen fin from dst
R - seen rst from src                  r - seen rst from dst
E - tcp conn established (SasA done)   T - tcp conn torn down (FafA done)

#SVS Domain 2007  Module   3
Proto SrcIP[:Port]          DstIP[:Port]          VLAN Action    Flags

#SVS Domain 2007  Module   4
Proto     SrcIP[:Port]        DstIP[:Port]       VLAN     Action    Flags
icmp      10.100.201.176      10.100.201.185     160      permit    PpOo
```

clear service-path connection

You can clear the connections (flow-table) maintained in the Cisco VSG by using the **clear service-path connection** command.

This example shows how to clear the flow-table connection output:

```
vsg# clear service-path connection
```

show vsg ip-binding

You can display a list of VM IP addresses and associated Virtual Network Service Profiles (VNSPs) with the associated policy set by using the **show vsg ip-binding** command. This information helps you to troubleshoot data path issues. The attribute manager determines which policy set to evaluate for a given packet (source IP address is the key for the lookup).

When debugging issues (for example, the wrong policy set or no policy), use this command to ensure that IP bindings (IP address to VNSP association) are correct. This association can also affect VNSP and VM attributes fetched by the attribute manager.

This example shows how to display the list of VM IP addresses and associated VNSPs:

```
vsn# show vsg ip-binding
------------------------------------------------------------------------------
   VM IP address          Security-Profile Name           Policy Name
------------------------------------------------------------------------------
 100.1.246.6        sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
 100.1.246.5        sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
 100.1.246.4        sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
 100.1.246.3        sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
 100.1.246.2        sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
 100.1.246.1        sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
 100.1.246.10       sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
 100.1.246.9        sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
 100.1.246.8        sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
 100.1.246.7        sec-profile-one@root/Tenant-one  policyset-one@root/Tenant-one
```

You can use the **show vsg ip-binding vm** command to show all IP VM bindings.

This example shows how to display all IP VM bindings:

```
vsg# show vsg ip-binding vm
------------------------------------------------------------------------------
   VM IP address               VM Name              Port Profile Name
------------------------------------------------------------------------------
 10.100.201.185            linux-206-185              profile_test2
 10.100.201.176            linux-206-176              profile_test2
```

You can use the show vsg ip-binding vm detail command to see more details about the IP VM bindings.

This example shows how to display the details of the IP VM bindings:

```
vsg# show vsg ip-binding vm detail
VM IP address    : 10.100.201.185
VM Name          : linux-206-185
VM uuid          : 421cefd6-29d1-4c8e-e563-2c3a4d58cd31
DV Port          : 1209::1c7b1c50-f1b7-9a71-259d-820f4713a4b1
Port Profile     : profile_test2

VM IP address    : 10.100.201.176
VM Name          : linux-206-176
VM uuid          : 421c44f9-91e0-b063-4fb0-ff3f4d736c3b
DV Port          : 1208::1c7b1c50-f1b7-9a71-259d-820f4713a4b1
```

```
Port Profile     : profile_test2
```

show vsg dvport {*dvport_id*}

You can display relevant information for a DV port by using the **show vsg dvport** command. A DV port is a logical representation of a vNIC. By default, this displays information for all DV ports. Specify a particular DV port with the *dvport id* parameter.

This example shows how to display the DV port information:

```
vsg# show vsg dvport

DV Port          : 576::bcaa1c50-8747-8d08-fe7e-a9aa8924bf8e
Security Profile : spcustom
VM uuid          : 421c5ae4-51c3-5dd9-60fa-a50cb04ed0ea
Port Profile     : vm_data
IP Addresses     :
   100.1.1.20
   100.1.1.10
```

show vsg vm

You can display information for all VMs on a VSG by using the **show vsg vm** command. In addition to the existing VM information, zone names are displayed in the command output. DV port information is not display to limit verbosity.

This example shows how to display all VMs on a VSG:

```
firewall-1# show vsg vm
VM uuid          : 42031129-65af-976b-5c5c-509966ffdede
VM attributes :
    name                      : gentoo-246-2
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.11
    cluster-name              :

VM uuid          : 4203326d-91d1-2fba-838a-3a551e5bcce1
VM attributes :
    name                      : gentoo-246-8
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.11
    cluster-name              :

VM uuid          : 420392dd-1146-f8eb-f0cb-363fb999a02d
VM attributes :
    name                      : gentoo-246-10
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.11
    cluster-name              :

VM uuid          : 42036819-f763-342a-8833-c24f9c55261f
VM attributes :
    name                      : gentoo-246-4
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
```

```
    host-name                 : 203.0.113.11
    cluster-name              :

VM uuid          : 420374a0-a81d-fe72-1dd8-f7b4ece9194c
VM attributes :
    name                      : gentoo-246-5
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.11
    cluster-name              :

VM uuid          : 4203625c-d9d0-1dde-228e-a2aaa97ad7c2
VM attributes :
    name                      : gentoo-246-1
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.11
    cluster-name              :

VM uuid          : 42034686-db79-478a-920f-2dd2cce07151
VM attributes :
    name                      : gentoo-246-7
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.11
    cluster-name              :

VM uuid          : 4203ac4a-a7f6-3320-436d-29a49c1c73e8
VM attributes :
    name                      : gentoo-246-9
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.11
    cluster-name              :

VM uuid          : 42033483-18b1-a89f-2f24-ae142365f061
VM attributes :
    name                      : gentoo-246-6
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.11
    cluster-name              :

VM uuid          : 420360fb-cfcc-21f0-b3dd-f3650ff37a6d
VM attributes :
    name                      : gentoo-246-3
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.11
    cluster-name              :
```

show vsg vm name {*name*}

You can display information of one or more VMs by entering the **show vsg vm name** command. The VM name should be specified as a parameter and can be the name or the first few characters of the name. The information for the VM includes the details of each DV port used by the VM and zones that the VM belongs to.

This example shows how to display information for the VM that has a name that starts with linux-204:

```
firewall-1# show vsg vm name linux-204
VM uuid          : 421ceac2-3b3f-67f9-b71c-3755d2c8cabe
VM attributes :
   cluster-name               : cluster23
   host-name                  : 203.0.113.11
   name                       : linux-204-184
   os-fullname                : red hat enterprise linux 4 (32-bit)
   os-hostname                :
   res-pool                   : resources
   tools-status               : not-installed
   vapp-name                  :
DV Port(s) :
   DV Port                    : 272::1c7b1c50-f1b7-9a71-259d-820f4713a4b1
   Security Profile           : SP-DC1@root/Cisco-Tenant1
   Port Profile               : profile_App2
   IP Addresses :
     20.100.201.184
   DV Port                    : 240::1c7b1c50-f1b7-9a71-259d-820f4713a4b1
   Security Profile           : SP-App1@root/Cisco-Tenant1
   Port Profile               : profile_App1
   IP Addresses :
     10.100.201.184
Zone(s) :
   zone_linux_204@root/Cisco-Tenant1
```

show vsg vm uuid {*vm_uuid*}

You can display relevant information for a particular VM by using the **show vsg vmuuid** command. The attribute manager looks up the VM attributes for the VM based on this association before doing a policy evaluation.

When debugging issues, such as when the wrong VM attributes are fetched, check the output of this command as well as the IP address to DV port mapping.

This example shows how to display the relevant information for a VM:

```
firewall-1# show vsg vm uuid
VM uuid          : 42031129-65af-976b-5c5c-509966ffdede
VM attributes :
    name                       : gentoo-246-2
    vapp-name                  :
    os-fullname                : other 2.6x linux (64-bit)
    tools-status               : not-installed
    host-name                  : 203.0.113.111
    cluster-name               :

VM uuid          : 4203326d-91d1-2fba-838a-3a551e5bcce1
VM attributes :
    name                       : gentoo-246-8
    vapp-name                  :
    os-fullname                : other 2.6x linux (64-bit)
    tools-status               : not-installed
    host-name                  : 203.0.113.111
```

```
    cluster-name                :

VM uuid          : 420392dd-1146-f8eb-f0cb-363fb999a02d
VM attributes :
    name                        : gentoo-246-10
    vapp-name                   :
    os-fullname                 : other 2.6x linux (64-bit)
    tools-status                : not-installed
    host-name                   : 203.0.113.111
    cluster-name                :

VM uuid          : 42036819-f763-342a-8833-c24f9c55261f
VM attributes :
    name                        : gentoo-246-4
    vapp-name                   :
    os-fullname                 : other 2.6x linux (64-bit)
    tools-status                : not-installed
    host-name                   : 203.0.113.111
    cluster-name                :

VM uuid          : 420374a0-a81d-fe72-1dd8-f7b4ece9194c
VM attributes :
    name                        : gentoo-246-5
    vapp-name                   :
    os-fullname                 : other 2.6x linux (64-bit)
    tools-status                : not-installed
    host-name                   : 203.0.113.111
    cluster-name                :

VM uuid          : 4203625c-d9d0-1dde-228e-a2aaa97ad7c2
VM attributes :
    name                        : gentoo-246-1
    vapp-name                   :
    os-fullname                 : other 2.6x linux (64-bit)
    tools-status                : not-installed
    host-name                   : 203.0.113.111
    cluster-name                :

VM uuid          : 42034686-db79-478a-920f-2dd2cce07151
VM attributes :
    name                        : gentoo-246-7
    vapp-name                   :
    os-fullname                 : other 2.6x linux (64-bit)
    tools-status                : not-installed
    host-name                   : 203.0.113.111
    cluster-name                :

VM uuid          : 4203ac4a-a7f6-3320-436d-29a49c1c73e8
VM attributes :
    name                        : gentoo-246-9
    vapp-name                   :
    os-fullname                 : other 2.6x linux (64-bit)
    tools-status                : not-installed
    host-name                   : 203.0.113.111
    cluster-name                :

VM uuid          : 42033483-18b1-a89f-2f24-ae142365f061
VM attributes :
    name                        : gentoo-246-6
    vapp-name                   :
    os-fullname                 : other 2.6x linux (64-bit)
    tools-status                : not-installed
    host-name                   : 203.0.113.111
    cluster-name                :
```

```
VM uuid          : 420360fb-cfcc-21f0-b3dd-f3650ff37a6d
VM attributes :
    name                      : gentoo-246-3
    vapp-name                 :
    os-fullname               : other 2.6x linux (64-bit)
    tools-status              : not-installed
    host-name                 : 203.0.113.111
    cluster-name              :
```

show vsg security-profile {[*vnsp-name*]| **detail** | **table**}

You can display information for a specific VNSP or all VNSPs by using the **show vsg security-profile** command. The attribute manager looks up custom attributes for a particular VNSP that is based on this association before doing a policy evaluation. By default, information is displayed for all VNSPs. You can specify a particular VNSP by using the *vnsp-name* argument.

When debugging issues such as the wrong policy set are evaluated, check if the correct policy set is associated with the VNSP. If custom attribute values are not correct, this command displays some details.

The detail version of this command includes names of the VMs that are using the security-profile in addition to their security-profile information. A VNSP name can be specified to get details of a specific security-profile.

This example shows how to display detailed information about a specific Cisco VSG security profile with the name sp_deny@root:

```
firewall-1# show vsg security-profile sp_deny@root detail
VNSP             : sp_deny@root
VNSP id          : 5
Policy Name      : ps_deny@root
Policy id        : 3
Custom attributes :
  Name           : vnsporg
  Value          : root
  Name           : profile1
  Value          : eng

Virtual Machines:
  sg-pg-vm206
  sg-pg-redhat
```

You can display the associated VNSP ID and policy for all VNSPs by using the **show vsg security-profile** command. The attribute manager uses this association when looking up a VNSP and associated policy from the packet that reaches the data0 interface of the Cisco VSG. When VPath redirects the packets to the Cisco VSG, the VNSP ID is added in the packet header.

This example shows how to display brief tabular information for the Cisco VSG security profile:

```
firewall-tenant-aa# show vsg security-profile table
------------------------------------------------------------------------------
         Security-Profile Name           VNSP ID      Policy Name
------------------------------------------------------------------------------
 default@root                             1       default@root
 sec-profile-AB@root/Tenant-A/Data-Center-B 30
 sec-profile-AA@root/Tenant-A/Data-Center-A 31
policyset-AA@root/Tenant-A/Data-Center-A
```

show vsg zone

You can display VM to zone mappings on a Cisco VSG by using the **show vsg zone** command.

This example shows how to display the VM to zone mappings on a Cisco VSG:

```
vsg# show vsg zone
Zone : zone2@root/test34
Virtual Machines :
linux-206-185
------------------------------------------------------------------------------
Zone : zone1@root/test34
Virtual Machines :
linux-206-176
------------------------------------------------------------------------------
```

show policy-engine stats

You can display statistics on the policy engine by using the **show policy-engine stats** command.

This example shows how to display the statistics for the Cisco VSG policy engine:

```
firewall-1# show policy-engine stats

Policy Match Stats:

default@root                 :         0
  default/default-rule@root  :         0 (Drop)
  NOT_APPLICABLE             :         0 (Drop)

policyset-one@root/Tenant-one          : 844935064
  policy-one/rule-z1@root/Tenant-one   : 808288619 (Permit)
  policy-one/rule-one@root/Tenant-one  :  36646445 (Permit)
  NOT_APPLICABLE                       :         0 (Drop)
```

This example shows how to use the help (?) feature of the command to display command options:

```
firewall-1# show policy-engine ?
  WORD   Enter policy-name to show its stats
  stats  Show the Stats

firewall-1# show policy-engine policyset-one@root/Tenant-one stats

Policy Match Stats:

policyset-one@root/Tenant-one          : 844935064
  policy-one/rule-z1@root/Tenant-one   : 808288619 (Permit)
  policy-one/rule-one@root/Tenant-one  :  36646445 (Permit)
  NOT_APPLICABLE                       :         0 (Drop)
```

clear policy-engine

You can clear the policy-engine statistics by using the **clear policy-engine** command.

This example shows how to see the options for clearing the policy-engine statistics:

```
firewall-1# clear policy-engine ?
  WORD   Enter policy-name to clear its stats
  stats  Clear the Stats
```

When the **stats** argument is used, the statistics are cleared and the only response for a successful action is a return to the prompt. This example shows how to clear the policy engine statistics:

```
firewall-1# clear policy-engine stats
```

show ac-driver statistics

You can display statistics that are collected in the AC driver module by using the **show ac-driver statistics** command. These statistics indicate how many packets are received, how many of those received packets are from vPath, how many packets are passed up to the service path, how many packets are passed as a response to the vPath and any error statistics, and so on.

This example shows how to display the AC driver module statistics:

```
firewall-1# show ac-driver statistics
#Packet Statistics:
  Rcvd Total                  852079858  Buffers in Use                  3190
  Rcvd VPath Pkts             848148272  Sent to VPath                846621771
  Sent to Service-Path        848148272  Sent to Control-Path           3931586
  All Drops                          0  Invalid LLC                          0
  Invalid OUI                        0  Invalid VNS Hdr                      0
  Invalid VNS PDU                    1  Service-Path not Inited              0
  Service-Path Down                  0  Rcvd Bad Descriptor                  0
  Send to Service-Path Err           0  Packet Offset Err                    0
  Send Bad Descriptor                0  Send NIC Err                         0
```

clear ac-driver statistics

You can clear the statistics that are collected in the AC driver module by using the **clear ac-driver statistics** command.

This example shows how to clear the statistics collected in the AC driver module:

```
vsg# clear ac-driver statistics
Cleared statistics successfully.
```

show system internal ac ipc-stats fe [*process-name*]

You can display internal statistics of the following processes by using the **show system internal ac ipc-stats fe** command:

- attribute-manager
- inspection-ftp
- inspection-rsh
- inspection-tftp
- service-path

This example shows how to display the statistics for the inspection-ftp process:

```
firewall-1# show system internal ac ipc-stats fe inspection-ftp
================================================================================
                        Instance:               1
                        IPC Type:             MTS(SAP 1326)
              Async requests sent:                0
         Async responses received:                0
          Async requests received:           764364
             Async responses sent:           764364
             Sendto requests sent:            32485
         Sendto requests received:            32485
                Async send errors:                0
             Async receive errors:                0
            Async response errors:                0
               Sendto send errors:                0
            Sendto receive errors:                0
```

```
                Receive errors:              0
                 Token errors :              0
    Destination not found errors:              0
          Sendto response errors:              0
                   Timer Errors :              0
                        Timouts :              0
                   Recv Queue Len:             11
               Queue Length High:              0
             Reciever Busy Errors:              0
================================================================================
```

clear system internal ac ipc-stats fe [*process-name*]

You can clear the internal statistics for the following processes by using the **clear system internal ac ipc-stats fe** command:

- attribute-manager
- inspection-ftp
- inspection-rsh
- inspection-tftp
- service-path

This example shows how to clear the statistics for the inspection-ftp process:

```
firewall-1# clear system internal ac ipc-stats fe inspection-ft
```

show inspect ftp statistics

You can display the following inspect FTP statistics pertaining to one vPath by using the **show inspect ftp statistics** command:

- The packets seen by the inspect FTP path from the vPath.
- Flows created by the inspect FTP path due to these packets.
- Packets dropped in the inspect FTP path due to various errors.

This example shows how to display the FTP statistics:

```
firewall-1# show inspect ftp statistics
Input packets                  764364
Dropped packets                     0
Reset-drop packets                  0
New connections                32485
Deleted connections            31064
IPC errors                          0
IPC allocation errors               0

SVS Domain   131  Module   4
Input packets                  764364
Dropped packets                     0
Reset-drop packets                  0
New connections                32485
Deleted connections            31064

firewall-1# show inspect ftp statistics svs-domain-id 131 module 4
Input packets                             764364
Dropped packets                                0
Reset-drop packets                             0
New connections                            32485
```

```
Deleted connections                     31064
Port zero drops                             0
Invalid port drops                          0
No port drops                               0
Port command long drops                     0
Rx port mismatch drops                      0
Command not port command drops              0
Embryonic connections                   32485
Embryonic connection failures               0
Memory allocations                      64970
Memory de-allocations                   63549
Memory allocation failures                  0
Command in reply mode drops                 0
Invalid command drops                       0
Un-supported command drops                  0
Command not terminated drops                0
Unexpected reply drops                      0
Command too short drops                     0
Reply code invalid drops                    0
Reply length negative drops                 0
Reply unexpected drops                      0
Rx command in command mode drops            0
```

clear inspect ftp statistics

Use the **clear inspect ftp statistics** command to clear the inspect FTP statistics globally when no option is given. When the SVS domain ID and the module are provided, the command clears the statistics of the specified module.

This command provides the following keyword filters:

- **svs-domain-id** *domain-id*—Displays only the Cisco VSG connections that are associated to the svs-domain specified in the *domain-id*.
- **module** *module-num*—Displays only the Cisco VSG connections that are associated to the SVS domain and VEM module specified in the *domain-id* and the *module-num*. Use this keyword filter only with the **svs-domain-id** filter.

This example shows how to clear the inspect FTP statistics:

```
firewall-1# clear inspect ftp statistics
firewall-1# clear inspect ftp statistics svs-domain-id 131 module 4
```

Troubleshooting Installation Issues

This chapter describes how to troubleshoot installation issues for the Cisco Virtual Security Gateway (VSG).

This chapter includes the following sections:

Verifying the VMware License Version

Before beginning to troubleshoot any installation issues, use this procedure to verify that your ESX server has the VMware Enterprise Plus license that includes the Distributed Virtual Switch feature.

BEFORE YOU BEGIN

Before beginning, you must know or do the following:

- Log in to the vSphere client where the Cisco VSG will be installed on the ESX server.
- Log in to the Cisco VSG CLI in EXEC mode.
- If your vSphere ESX server does not have the Enterprise Plus license, you must upgrade your license.

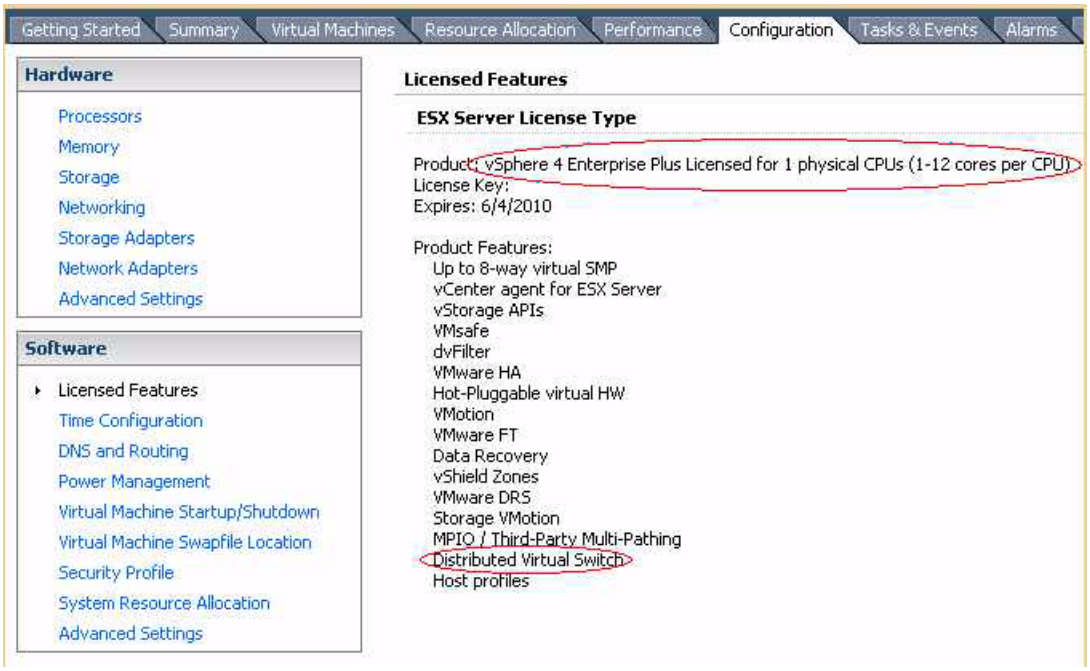
DETAILED STEPS

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

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

The Enterprise Plus licensed features appear. See Figure 3-1.

Figure 3-1 Verification of License



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 the ESX server has an Enterprise Plus license, you do not have to do anything because the Cisco VSG is available to you.
- If the ESX server does not have an Enterprise Plus license, upgrade the VMware License to an Enterprise Plus license so that you can see the Cisco VSG.

Verifying Port Group Assignments for a Cisco VSG VM Virtual Interface

Create the following port profiles on the VSM:

- Data interface port profile (VLAN is the data VLAN)
- HA interface port profile (VLAN is the HA VLAN)
- Management port profile (VLAN is the management VLAN)

Ensure that the port groups are assigned to the three virtual interfaces of the Cisco VSG VM in the following order:

1. Network adapter 1 for the data port group
2. Network adapter 2 for the management port group
3. Network adapter 3 for the HA port group

The Cisco VSG VM network adapter 1, network adapter 2, and network adapter 3 are carrying the data VLAN, the HA VLAN, and the management VLAN respectively.

OVA Installation Behavior

During OVA installation, the following error message might appear:

```
"The network card VirtualE1000 has dvPort backing, which is not supported. This could be
because the host does not support vDS, or because the host is not using vDS."
```

To work around this error, ensure that all three network interfaces in the Cisco VSG port profile are set to the VM Network (port profile from vSwitch) during OVA installation.

Once the virtual machine is created, the port profile for the three interfaces should be changed according to the *Cisco VSG for VMware vSphere and Cisco PNSC Installation and Upgrade Guide* for your release number.

Troubleshooting Licensing Issues

This chapter describes how to troubleshoot issues that are related to firewall licensing on the Virtual Supervisor Module (VSM).

This chapter includes the following sections:

Information About Licensing

CISCO VSG follows universal licensing mode in which if VEM licenses are there, two VSG licenses are given.

A module is licensed or unlicensed according to the following definitions:

- Firewalled module—A VEM is considered to be firewalled if it can acquire licenses for all of its CPU sockets.
- Nonfirewalled module—A VEM is considered to be nonfirewalled if it cannot acquire licenses for any, or a subset of, its CPU sockets.

If a VEM is nonfirewalled, all the virtual Ethernet ports on the VEM that correspond to the virtual machines (VMs) are kept in pass-through mode, so that these virtual machines are not firewalled.

By default, VSG allocate 2 licenses for each VEM.

For additional information about licensing, see the *Cisco Nexus 1000V for KVM License Configuration Guide*.

Troubleshooting License Installation Issues

This section describes how to troubleshoot Cisco VSG license installation issues.

Note This section assumes that you have a valid Cisco VSG license file.

For additional information about licensing, see the *Cisco Nexus 1000V for KVM License Configuration Guide*.

This section includes the following topics:

License Troubleshooting Checklist

Before you start the troubleshooting process, follow these requirements:

- Make sure that the name of the license file is less than 32 characters.
- Make sure that no other license file with the same name is installed on the VSM. 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 Technical Assistance Center (TAC) Team.
- Make sure that the host ID in the license file is the same as the host ID on the switch.

Removing an Evaluation License File

If an evaluation license file is already installed on the VSM, you must remove it from the VSM before installing a permanent license file. For more information, see the *Cisco Nexus 1000V for KVM License Configuration Guide* for your release number.

Determining Cisco VSG License Usage

You can view the Cisco VSG license state of the VEMs on your VSM and the number of CPU sockets per VEM by entering the **module vem 3 execute vemcmd show vsn config** command.

This example shows how to confirm the Cisco VSG license state:

```
vsm# module vem 3 execute vemcmd show vsn config
  VNS Enabled  | VNS Licenses Available   2
  VSN#  VLAN              IP          STATIC-MAC        LEARNED-MAC  LTLs
     1   754      200.1.1.10  00:00:00:00:00:00  00:50:56:83:00:01     0
```

In this command output, VEM 3 is licensed. It has two CPU sockets and it currently uses two firewall licenses.

Viewing Installed License Information

You can view the installed license count by entering the **show license usage** command.

This example shows how to display the installed licenses count:

```
VSM-(config)# show license usage
Feature                      Ver Ins Lic   Status Expiry Date Comments
                                     Count
--------------------------------------------------------------------------------
```

```
NEXUS1000V_STINGRAY_PKG        1.0 No     0  Unused             -
NEXUS1000V_LAN_SERVICES_PKG    3.0 No  1024  In use 04 Oct 2014 -
NEXUS_ASA1000V_SERVICES_PKG    1.0 No   512  Unused 04 Oct 2014 -
NEXUS1000V_INTERCLOUD_VM_PKG   1.0 No    16  Unused 04 Oct 2014 -
--------------------------------------------------------------------------------
Note: Licenses are not required for Essential Edition
--------------------------------------------------------------------------------
```

Troubleshooting Module Issues

This chapter describes how to troubleshoot various issues that could occur while the Cisco VSG is communicating with the Virtual Supervisor Module (VSM), Virtual Ethernet Module (VEM), Cisco Virtual Network Management Center (PNSC), or the vCenter Server.

This chapter includes the following sections:

Troubleshooting Cisco VSG and VSM Interactions

This section describes how to troubleshoot issues with the Cisco VSG and VSM interactions.

The port profile used to bring up the data interface of the Cisco VSG should not have any vn service or org configured.

This example shows how to use a port profile to bring up the Cisco VSG data interface:

```
vsm# show port-profile name vsg-data
port-profile vsg-data
 type: Vethernet
 description:
 status: enabled
 max-ports: 32
 inherit:
 config attributes:
  switchport mode access
  switchport access vlan 754
  no shutdown
 evaluated config attributes:
  switchport mode access
  switchport access vlan 754
  no shutdown
 assigned interfaces:
  Vethernet4
  Vethernet6
 port-group: vsg-data
```

```
 system vlans: none
 capability l3control: no
 capability iscsi-multipath: no
 port-profile role: none
 port-binding: static
```

Make sure that you add the Cisco VSG service VLAN and HA VLAN as part of the allowed VLAN under the uplink port profile. Without adding this information into the allowed VLAN, Cisco VSGs may not pair. If you have a Cisco VSG on one VEM and the VMs to be firewalled are on another VEM, you must make sure that the Cisco VSG service VLAN is added as the allowed VLAN under the uplink port profile.

The example shows that VLAN 753 and 754 are added as part of the trunk. The VLAN 751 is used for control (VSM), the VLAN 752 for packet, the VLAN 754 for the Cisco VSG service, and the VLAN 753 for the Cisco VSG high availability.

```
vsm# show port-profile name perf-uplink
port-profile perf-uplink
 type: Ethernet
 description:
 status: enabled
 max-ports: 32
 inherit:
 config attributes:
  switchport mode trunk
  switchport trunk allowed vlan 751-754
  no shutdown
 evaluated config attributes:
  switchport mode trunk
  switchport trunk allowed vlan 751-754
  no shutdown
 assigned interfaces:
  Ethernet3/4
  Ethernet4/4
 port-group: perf-uplink
 system vlans: 751-752
 capability l3control: no
 capability iscsi-multipath: no
 port-profile role: none
 port-binding: static
```

For the port profiles that are used to protect the VMs, make sure that you provide the correct vn service IP (the exact data 0 IP address of the Cisco VSG), and the service VLAN and the security profile name. Make sure under the org that you have configured the tenant name as root/Tenant-cisco.

Troubleshooting Cisco VSG and VEM Interactions

This section describes how to troubleshoot issues with Cisco VSG and VEM interactions.

This section includes the following topics:

Policies Configured on the Cisco VSG but Not Effective

Sometimes, when the policies are configured on the Cisco VSG and the data traffic is sent from the VMs, traffic flows through the Cisco Nexus 1000V switch as if the firewall service is not enabled on the port.

Possible reasons:

- VMs are not bound to the proper port profiles.
- The license is not available or is not installed/configured on the module.

Verifications:

- Check if the VMs to be protected are bound to proper port profiles. The port profiles are expected to have the org/vn-service identified.
- On the Cisco VSG, enter the **show vsg ip-binding** command to see if the VM IP to service profile binding is present.
- On the VEM, enter the **vemcmd show vsn binding** command to check if the VM is protected by the firewall.
- To get the lower threshold limit (LTL) of the VM on the VEM, enter the **vemcmd show port** command as follows:

```
vem# vemcmd show port | grep w2k-client_110.eth2 <--- VM name
50 Veth5 UP UP FWD 0 w2k-client_110.eth2
```

Verify if the LTL is found as follows:

```
vem# vemcmd show vsn binding
VSG Services Enabled  | VSG Licenses Available   2 <--- should be nonzero
ASA Services Disabled | ASA Licenses Available   0
LTL  PATH VSN  SWBD    IP      P-TYPE  P-ID
50     1   1   101  10.1.1.230    1       3
```

The VSG Licenses Available message should display a nonzero value in the output.

Note All **vemcmd** commands can be executed by logging into the ESX via SSH.

Traffic Fails to Reach Destination with a Permit Policy Configured on the Cisco VSG

When policies are configured on the Cisco VSG to permit a certain type of traffic, but the traffic does not reach the destination, a complete failure can result.

Possible reason:

The Virtual Ethernet Modules (VEMs) have not learned the MAC address of the Cisco VSG.

Verifications:

Check if the Cisco VSG MAC address is learned on all the VEMs that host the protected VMs involved in the communication by entering the **vemcmd show vsn config** command on the VEM.

This example shows how to display the Cisco VSG configuration:

```
vem# vemcmd show vsn config
VSG Services Enabled | VNS Licenses Available 2
ASA Services Disabled| ASA Licenses Available 0
VSN#  SWBD IP         MAC               LTLs   VER  VER-BITMAP
```

```
1    101   10.1.1.230 00:50:56:be:4f:c6     1       2   1,2
```

The following conditions should be displayed on the command output:

- The VNS Licenses Available message should display a nonzero value.
- The learned MAC address in the above output should not be 00:00:00:00:00:00.
- The learned MAC address should match with the MAC address of the Cisco VSG that is intended to protect the VMs.

You can find the MAC address of the Cisco VSG by entering the **show interface data 0** command.

This example shows how to display information on the interface for the Cisco VSG:

```
vsg# show interface data 0
data0 is up
Hardware: Ethernet, address: 0050.569c.3cc5 (bia 0050.569c.3cc5) <----
MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA
full-duplex, 10 Gb/s
Auto-Negotiation is turned on
Rx
0 input packets
Tx
8084 output packets
```

If the learned MAC address in the **vemcmd show vsn config** command is 00:00:00:00:00:00, manually check if the Cisco VSG service (data) interface is bound to the proper port profile and has the right VLAN configured.

You can check the Cisco VSG service interface assignment on the VEM by entering the **vemcmd show** command.

This example shows how to check the Cisco VSG service interface assignment on the VEM:

```
vem# vemcmd show vlan 501         <------------- 501 is the service VLAN
VLAN 501, vdc 1, swbd 501, hwbd 11, 5 ports
Portlist:
6 vns
18 vmnic1
58 tenant1-primary ethernet0      <------------- Cisco VSG VM name
```

The Cisco VSG VM name should be displayed as part of the output.

You can display the port profile that is associated with the Cisco VSG's service interface by entering the **show port-profile name** *pp-name* command on the VSM.

If the Cisco VSG is bound to the proper port profile and has the correct service VLAN, check the upstream switches. Ensure that this service VLAN is configured across all ports in all upstream switches to which all the VEMs (those talking to that Cisco VSG) are connected.

You can ensure that the service VLAN is configured and enabled (active) on the VSM by entering the **show vlan** command.

This example shows how to display the VLAN configurations:

```
vsm# show vlan

VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
1    default                           active
501  VLAN0501                          active    Po1, Po2, Po3, Po4, Veth3
```

Make sure that the following occurs:

- Service VLAN (501) is configured in the uplink port profile on the VSM.
- Service VLAN is not configured as a system VLAN on the uplink port profile.

You can confirm the configuration by entering the **show running-config port-profile system-data-uplink** command.

This example shows how to confirm the configuration:

```
vsm# show running-config port-profile system-data-uplink

!Command: show running-config port-profile system-data-uplink
!Time: Thu Feb 24 13:06:30 2011

version 4.2(1)SV1(4)
port-profile type ethernet system-data-uplink
  vmware port-group
  switchport mode trunk
  switchport trunk allowed vlan 51-53,501
  no shutdown
  system vlan 51-52
  state enabled
```

Security Posture Not Maintained After the VMotion of the VM to the new ESX Host

After performing VMotion of the traffic VM, the security posture as defined by the policies in the Cisco VSG can be disrupted.

Possible reasons:

- The license was not checked out on the new module.
- The VEM did not learn the MAC address of the Cisco VSG.

Verifications:

- Check if the Cisco VSG MAC is learned on all the VEMs that host the protected VMs involved in communication by entering the **vemcmd show vsn config** command.

  This example shows how to display the Cisco VSG MAC information:

```
vem# vemcmd show vsn config
VSG Services Enabled  | VSG Licenses Available   2
ASA Services Enabled  | ASA Licenses Available   2
    VSN#  SWBD              IP               MAC  LTLs VER     VER-BITMAP
      2  3756     10.10.10.202  00:50:56:83:00:1e     2   2           1,2
     27  3770       172.31.2.1  00:50:56:a4:0f:36     1   2           1,2
     28  3756     10.10.11.202  00:50:56:a4:0f:3d     1   2           1,2
```

  - The VNS Licenses Available message should display a nonzero value.
  - The learned MAC address should not be 00:00:00:00:00:00 for the layer 2 adjacent node.
  - The learned MAC address should match with the MAC address of the Cisco VSG that is intended to protect the VMs.

  This example shows how to find the MAC address of the Cisco VSG on the corresponding Cisco VSG:

```
vsg# show interface data 0
data0 is up
  Hardware: Ethernet, address: 0050.569c.3cc5 (bia 0050.569c.3cc5) <----
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
```

```
   reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA
full-duplex, 10 Gb/s
Auto-Negotiation is turned on
Rx
  0 input packets
Tx
  8084 output packets
```

- If the learned MAC address in the **vemcmd show vsn config** command is 00:00:00:00:00:00, manually check if the Cisco VSG service (data) interface is bound to the proper port profile and has the right VLAN configured.

  This example shows how to check the Cisco VSG service interface assignment on the VEM:

```
vem# vemcmd show vlan 501         <------------- 501 is the service VLAN
VLAN 501, vdc 1, swbd 501, hwbd 11, 5 ports
Portlist:
6 vns
18 vmnic1
58 tenant1-primary ethernet0      <------------- Cisco VSG VM name
```

The Cisco VSG VM name should be displayed as part of the output.

You can view the port-profile information for the Cisco VSG's service interface by entering the **show port-profile name** *pp-name* command on the VSM.

If the Cisco VSG is bound to the proper port profile and has the correct service VLAN, check the upstream switches. Ensure the service VLAN is configured across all ports in all upstream switches to which all the VEMs (those talking to that Cisco VSG) are connected.

Policy Decision Inconsistent with the Port Profile Changes

When policy decisions are inconsistent with the port-profile changes, either of these conditions can exist:

- A user changed the port profile of the traffic VM from one Cisco VSG port profile to another (having a different security profile).
- A policy is modified and the newer policy does not take immediate effect.

Reason:

Because of the existing flows, the old policy decision is continued.

Action:

Administrators must clear the flows in the vPath and Cisco VSG when the policy is modified.

Using vPath Ping to Determine Connectivity

You can use the vPath **ping** command to determine the connectivity between the Cisco VSG and the VEM.

This example shows how to ping the Cisco VSG connections and if they are reachable:

```
VSM-1# ping vservice node all src-module all
ping vsn 106.1.1.1 vlan 54 from module 3 5, seq=0 timeout=1-sec
  module(usec)   :  3(156)  5(160)
ping vsn 110.1.1.1 vlan 54 from module 3 5, seq=0 timeout=1-sec
  module(failed) :  3(VSN ARP not resolved)  5(VSN ARP not resolved)
```

```
ping vsn 106.1.1.1 vlan 54 from module 3 5, seq=1 timeout=1-sec
  module(usec)   :  3(230)  5(151)
ping vsn 110.1.1.1 vlan 54 from module 3 5, seq=1 timeout=1-sec
  module(failed) :  3(VSN ARP not resolved)  5(VSN ARP not resolved)

ping vsn 106.1.1.1 vlan 54 from module 3 5, seq=2 timeout=1-sec
  module(usec)   :  3(239)  5(131)
ping vsn 110.1.1.1 vlan 54 from module 3 5, seq=2 timeout=1-sec
  module(failed) :  3(VSN ARP not resolved)  5(VSN ARP not resolved)

ping vsn 106.1.1.1 vlan 54 from module 3 5, seq=3 timeout=1-sec
  module(usec)   :  3(248)  5(153)
ping vsn 110.1.1.1 vlan 54 from module 3 5, seq=3 timeout=1-sec
  module(failed) :  3(VSN ARP not resolved)  5(VSN ARP not resolved)

ping vsn 106.1.1.1 vlan 54 from module 3 5, seq=4 timeout=1-sec
  module(usec)   :  3(259)  5(126)
ping vsn 110.1.1.1 vlan 54 from module 3 5, seq=4 timeout=1-sec
  module(failed) :  3(VSN ARP not resolved)  5(VSN ARP not resolved)
```

This example shows how to display vService ping options:

```
VSM-338-STRESS# ping vservice node ?
all All vServices created
ip IP Address
vlan VLAN Number
vxlan VXLAN
```

This example shows how to display vService ping options for all source modules:

```
VSM-338-STRESS# ping vservice node all src-module ?
<3-258> Module number
all All modules in VSM
vpath-all All modules having VMs associated to vServices
```

This example shows how to set up a ping for all source modules from a specified IP address:

```
VSM-338-STRESS# ping vservice node ip 11.11.11.72 src-module all

ping vsn 11.11.11.72 vlan 0 from module 3 4 5 6 7, seq=0 timeout=1-sec
    module(usec) : 3(707) 4(681) 5(748) 6(759) 7(441)

ping vsn 11.11.11.72 vlan 0 from module 3 4 5 6 7, seq=1 timeout=1-sec
    module(usec) : 3(598) 4(362) 5(642) 6(534) 7(574)

ping vsn 11.11.11.72 vlan 0 from module 3 4 5 6 7, seq=2 timeout=1-sec
    module(usec) : 3(525) 4(289) 5(486) 6(480) 7(568)

ping vsn 11.11.11.72 vlan 0 from module 3 4 5 6 7, seq=3 timeout=1-sec
    module(usec) : 3(628) 4(389) 5(585) 6(436) 7(621)

ping vsn 11.11.11.72 vlan 0 from module 3 4 5 6 7, seq=4 timeout=1-sec
    module(usec) : 3(698) 4(244) 5(506) 6(620) 7(514)
```

This example shows to set up a ping for all vPath source modules for a specified IP address:

```
VSM-338-STRESS# ping vservice node ip 11.11.11.72 src-module vpath-all
ping vsn 11.11.11.72 vlan 0 from module 3 4 5 6 7, seq=0 timeout=1-sec
    module(usec) : 3(594) 4(403) 5(618) 6(546) 7(564)

ping vsn 11.11.11.72 vlan 0 from module 3 4 5 6 7, seq=1 timeout=1-sec
    module(usec) : 3(651) 4(415) 5(616) 6(517) 7(498)

ping vsn 11.11.11.72 vlan 0 from module 3 4 5 6 7, seq=2 timeout=1-sec
```

```
    module(usec) : 3(624) 4(385) 5(482) 6(533) 7(530)
```

This example shows how to set up a ping for all source modules of a specified IP address with a time-out and a count:

```
VSM-1# ping vservice node all src-module all timeout 2 count 3
ping vsn 10.1.1.60 vlan 501 from module 4 5 7, seq=0 timeout=2-sec
  module(usec)   :  4(444)  5(238)  7(394)

ping vsn 10.1.1.60 vlan 501 from module 4 5 7, seq=1 timeout=2-sec
  module(usec)   :  4(259)  5(154)  7(225)

ping vsn 10.1.1.60 vlan 501 from module 4 5 7, seq=2 timeout=2-sec
```

module(usec) : 4(227) 5(184) 7(216)

Troubleshooting VSM and Cisco PNSC Interactions

After registering the VSM to the Cisco PNSC, you can check the status of the VSM and Cisco PNSC policy agents by entering the **show nsc-pa status** command.

This example shows how to check the status:

```
vsm# show nsc-pa status
NSC Policy-Agent status is - Installed Successfully.
```

If there is a failure, there can be several reasons. One failure could be because the Cisco PNSC is unreachable or dead. Ping to the Cisco PNSC IP to check for a response. If there is no response, look at the network connectivity.

Another reason could occur because of the wrong shared secret.

This example shows the results of this type of failure:

```
vsm# show nsc-pa status
NSC Policy-Agent status is - Installation Failure
Incorrect shared secret.
```

Provide the correct password and register again.

On the Cisco PNSC GUI, on the Administration > Service Registry > Clients tab, make sure that the registered VSM is shown as registered under the Oper State column.

On the Cisco PNSC GUI, make sure that the org is configured in the same way as in the port profile. The registered VSM should also be available under the Resources > Virtual Supervisor Modules. If the org is not properly configured on the port profile, the Config State will display as “org-not-found” under the port profiles tab of the registered VSM. After editing the port profile with the correct org name, the Config State changes to OK.

Troubleshooting Cisco VSG and Cisco PNSC Interactions

After registering the Cisco VSG to the Cisco PNSC, you can check the status by entering the **show nsc-pa status** command.

This example shows how to check the Cisco VSG registration status:

```
vsg# show nsc-pa status
NSC Policy-Agent status is - Installed Successfully.
```

If there is a failure, there can be several reasons. One failure could be because the Cisco PNSC is unreachable or dead. Ping to the Cisco PNSC IP to check for a response. If there is no response, look at the network connectivity.

Another reason could occur because of the wrong shared secret.

This example shows how to display the results of this type of failure:

```
vsg# show nsc-pa status
NSC Policy-Agent status is - Installation Failure
Incorrect shared secret.
```

Provide the correct password and register again.

On the Cisco PNSC GUI, on the Administration > Service Registry > Clients tab, make sure that the registered VSM is shown as registered under the Oper State column.

Troubleshooting Cisco PNSC and vCenter Server Interactions

To enable the Cisco PNSC to communicate with the vCenter Server, you must have installed the Cisco PNSC's vCenter extension XML plug-in.

The vCenter Server is added to the Cisco PNSC with the provided IP address and name under Administration > VM Managers > Add VM manager. The Operational State of the newly added vCenter Server indicates that it is up.

Other possible operational states could be unreachable or bad credentials. If the state is unreachable, the vCenter Server is down or could not be reached. To check if you can access the vCenter server on the Cisco PNSC, use SSH to the Cisco PNSC with the user as admin and the PNSC password.

You can check reachability by entering the **connect local-mgmt** command.

This example shows how to access the vCenter Server:

```
pnsc# connect local-mgmt
Cisco Virtual Network Management Center
TAC support: http://www.cisco.com/tac
Copyright (c) 2002-2011, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained in this software are
owned by other third parties and used and distributed under
license. Certain components of this software are licensed under
the GNU General Public License (GPL) version 2.0 or the GNU
Lesser General Public License (LGPL) Version 2.1. A copy of each
such license is available at
http://www.opensource.org/licenses/gpl-2.0.php and
http://www.opensource.org/licenses/lgpl-2.1.php
```

Use the **ping** command to check if you can reach the vCenter Server (assuming that the vCenter Server does not block the **ping** command).

On the Cisco PNSC GUI, go to Administration > VMManagers tab and expand the VM Managers. Click on the vCenter Server object and review the right pane. If the state shows as bad-credentials, you have not registered the vCenter Server extension XML plugin for that vCenter Server. Go to the vCenter Server that is being added and install the vCenter Server extension XML plugin. For instructions, see "Chapter 7 - Configuring VM Managers" of the *Cisco Virtual Network Management Center GUI Configuration Guide*.

Troubleshooting the Cisco VSG and VEM Interactions When the Cisco VSG is on a VXLAN in a Service-Chain

You can run a series of checks to ensure that interactions between the Cisco VSG and VEM are seamless.

Run the following verifications:

- Check if the Cisco VSG is alive by using the **show vservice brief** command from the VSM.

  This example shows how to display the Cisco VSG configuration:

```
vsm# show vservice brief
#License Information
Type      In-Use
vsg         7
asa         7

#Node Information
ID  Name                        Type   IP-Address      Mode   State   Module
1   node_10.1.1.40_l3_fclose    vsg    10.1.1.40        l3     Unreach 3,9,
3   node_10.1.1.40_501_fclose   vsg    10.1.1.40        vxlan  Alive   4,9,11,
5   node_10.1.1.45_502_fclose vsg    10.1.1.45        vxlan  Unreach 9,
9   VASA1                       asa    192.168.200.221 v-53   Alive   3,9,11,
13  VASA-vxlan-222              asa    192.168.200.222 vxlan  Alive   4,9,11,
16  EL1                         vsg    7.1.1.1         v-501  Unreach 4,
17  EL2                         vsg    7.1.1.1         v-502  Unreach 3,
```

  If a specific Cisco VSG is not alive (wherein ‘Unreach’ or ‘??’ is displayed), use the **show vservice detail node_ipaddr** *node ip* command for further analysis.

- Check if the Cisco VSG node definition has “adjacency l2 vxlan” in the output. For example:

```
vservice node node_10.1.1.40_501_fclose type vsg
  ip address 10.1.1.40
  adjacency l2 vxlan bridge-domain segment2
  fail-mode close
```

- Check the port profile attached to the VM. It should be pointing to either the vsevice node VSG directly or to a service path that contains the corresponding Cisco VSG.
- Check the port profile attached to the Cisco VSG data interface, which should be on a bridge domain. It must not have any org/vservice configuration. For example:

```
port-profile type vethernet segment-5001-nofw
  vmware port-group
  switchport mode access
  switchport access bridge-domain segment2
  no shutdown
  state enabled
```

Troubleshooting Policy Engine Issues

This chapter describes how to troubleshoot issues that might occur on the policy engine.

This chapter includes the following sections:

Policy Engine Troubleshooting Commands

When there are policy engine issues, use these commands to troubleshoot:

- **show run rule**—Displays all rules configured in the Cisco VSG
- **show run policy**—Displays all policies configured in the Cisco VSG
- **show run zone**—Displays all zones configured in the Cisco VSG
- **show run object-group**—Displays all object-groups configured in the Cisco VSG
- **show policy-engine stats**—Displays statistics about the rule hits in the Cisco VSG
- **clear policy-engine stats**—Clears the statistics about the rule hits in the Cisco VSG

Policy/Rule Not Working as Expected

When policies or rules do not work as expected, do the following:

- Check the show policy-engine statistics and verify that the hits are incrementing by entering the **show policy-engine stats** command. If not, go to the module interactions section to see why hits are not incrementing.
- When policy engine statistics are incrementing, check the rule name that is being hit.
- View the configuration of the rule by entering the **show run rule** *rule-name* command. Verify that the conditions are configured correctly.

Policy/Rule Based on VM Attributes Not Working - But Without VM Attributes Policy/Rule Works

A policy or rule with VM attributes requires additional data for the Cisco VSG to evaluate the policy engine. This data, if not complete, can result in incorrect or not applicable hits in the statistics. When the policy or rule is configured with VM attributes, make sure that you see VM information in the following outputs:

- **show vsg ip-binding**—The output should have the IPs of all the VMs for which the rules will be written in the Cisco VSG.
- **show vsg dvport**—The output should have the port profile and IP information of all the VMs for which rules will be written in the Cisco VSG.
- **show vsg vm**—The output should have VM attribute values (whichever is present in the vCenter for a given VM) of all the VMs for which rules will be written in the Cisco VSG.

Policy/Rule Configured for Non-Firewalled VMs (port profiles) Not Working

To enable firewall protection for a VM, you must configure the vn-service and org CLI in the port profile at the VSM—this enables access to IP addresses and other attributes for the VM.

To write policies or rules for VMs based on the vCenter attributes (and at the same time not be protected), configure the org CLI only in the port profile to enable learning of IP addresses and other attributes for the VM with no firewall protection (for example, a client VM running Windows OS and a server running the Linux OS). To turn on firewall protection for the server VM (any traffic to or from server VM is protected by the Cisco VSG but not the client VM), write a rule saying that the source with the Windows OS and destination with the Linux OS VM is permitted by doing the following:

- Configure the vn-service and org CLI in the server VM port profile at the VSM.
- Configure the org CLI for the client VM port profile at VSM (no vservice).
- Write a rule with a source condition OS name that contains the Windows and a destination VM name server VM, action permit.

Policy Engine Statistics Show Hits as 0 and Traffic Not Reaching the Cisco VSG

Verify if the correct MAC address is displayed by entering the **show vservice brief** in the VSM. The MAC address should be the MAC address of the Cisco VSG data interface.

This example shows the **show vservice brief** output:

```
VSM-338-STRESS# show vservice brief
--------------------------------------------------
License Information
--------------------------------------------------
Type In-Use-Lic-Count UnLicensed-Mod
asa 2

--------------------------------------------------
```

```
Node Information
-------------------------------------------------
ID Name Type IP-Address Mode MTU State Module
1 ASA asa 33.33.33.34 v-756 NA Alive 6,
2 VPX adc 11.11.11.194 l3 1500 Alive 3,4,5,6,7,
3 VSG vsg 11.11.11.72 l3 NA Alive 3,4,5,6,7,
5 VWAAS vwaas 44.44.44.100 v-757 1500 Alive 5
```

If the MAC address is correct, check the following:

- Confirm that the buffers in use are not zero by entering the **show ac-driver statistics** command. If zero, check/fix the adapter type.
- The Cisco VSG data0 interface's adapter type in the VSM VM properties should be set to VMXNET3.
- If the Cisco VSG data interface adapter type E1000 does not work properly, set to VMXNET3.

When the Cisco VSG is deployed using the OVA format, the Cisco VSG does not have this issue because the adapter type is automatically correctly selected.

Troubleshooting High Availability Issues

This chapter describes how to troubleshoot issues related to high availability (HA).

This chapter includes the following sections:

Information About Cisco VSG High Availability

Cisco VSG high availability (HA) is a subset of the Cisco NX-OS HA. The following HA features minimize or prevent traffic disruption in the event of a failure:

Redundancy

Cisco VSG redundancy is equivalent to HA pairing. The two possible redundancy states are **active** and **standby**. An active Cisco VSG is always paired with a standby Cisco VSG. HA pairing is based on the Cisco VSG ID. Two Cisco VSGs that are assigned an identical ID are automatically paired. All processes running in the Cisco VSG are data path critical. If one process fails in an active Cisco VSG, a failover to the standby Cisco VSG occurs instantly and automatically.

Isolation of Processes

The Cisco VSG software contains independent processes known as services. These services perform a function or set of functions for a subsystem or feature set of the Cisco VSG. Each service and service instance runs as an independent, protected process. This operational process ensures a highly fault-tolerant software infrastructure and fault isolation between services. Any failure in a service instance does not affect any other services running at that time. Additionally, each instance of a service runs as an independent process. For example, two instances of a routing protocol run as separate processes.

Cisco VSG Failovers

The Cisco VSG HA pair configuration allows uninterrupted traffic forwarding by using stateful failovers when a failure occurs.

Problems with High Availability

The following key problems are found in Cisco VSG HA. In addition to these issues, some of the common Cisco NX-OS HA symptoms are listed in Table 7-1. The symptoms that are related to high availability, their possible causes, and recommended solutions are as follows.

- Cisco VSG pair communication problems
- Cisco VSGs do not reach an active/standby status
- Sometimes, the Cisco VSG reboots continuously when tenants share the management network (for example, collisions of the domain ID):
  - In a multitenant environment, if there is a shared management network and a collision occurs in the domain ID (two or more tenants using the same domain ID) spontaneous reboots of the Cisco VSGs that are involved in the collision are triggered. There is no workaround for this issue. When domain IDs are provisioned, they must be unique across all tenants
- Cisco VSGs in the HA pair get stuck in bash# prompt mode during reboots/upgrades/switchovers
- Cisco VSGs in the HA pair get stuck in boot# prompt mode during reboots/upgrades/switchovers

Table 7-1 Problems with High Availability

Symptom	Possible Cause	Solution
The active Cisco VSG does not see the standby Cisco VSG.	Roles are not configured properly: • primary • secondary	Do the following: 1. Verify the role of each Cisco VSG by entering the **show system redundancy status** command. 2. Update an incorrect role by entering the **system redundancy role** command. 3. Save the configuration by entering the **copy run start** command.
	Network connectivity problems are occurring between the Cisco VSG and the upstream and virtual switches. The problem could be in the control or management VLAN.	Restore connectivity as follows: 1. From the vSphere client, shut down the Cisco VSG, which should be in standby mode. 2. From the vSphere client, bring up the standby Cisco VSG after network connectivity is restored.

Table 7-1 Problems with High Availability (continued)

Symptom	Possible Cause	Solution
The active Cisco VSG does not complete synchronization with the standby Cisco VSG.	A version mismatch between Cisco VSGs might be occurring.	Do the following: 1. Verify the software version on both Cisco VSGs by entering the **show version** command. 2. Reinstall the secondary Cisco VSG with the same version used in the primary.
	Fatal errors occur during the gsync process. Check the gsyncctrl log by entering the **show system internal log sysmgr gsyncctrl** command and look for fatal errors.	Reload the standby Cisco VSG by entering the **reload module** *standby_module_number* command. See the "Reloading a Module" section on page 7-8.
The standby Cisco VSG reboots periodically.	The Cisco VSG has connectivity only through the management interface. When a Cisco VSG is able to communicate through the management interface, but not through the control interface, the active Cisco VSG resets the standby to prevent the two Cisco VSGs from being in HA mode and out of sync.	Check control VLAN connectivity between the primary and secondary Cisco VSG by entering the **show system internal redundancy info** command. In the output, **degraded_mode flag** = true. If there is no connectivity, restore it through the control interface.
Both Cisco VSGs are in active mode.	The following network connectivity problems might be occurring: • Check for control and management VLAN connectivity between the Cisco VSG at the upstream and virtual switches. • When the Cisco VSG cannot communicate through any of these two interfaces, they both try to become active.	If network problems exist, do the following: 1. From the vSphere client, shut down the Cisco VSG, which should be in standby mode. 2. From the vSphere client, bring up the standby Cisco VSG after network connectivity is restored.

Table 7-1 Problems with High Availability (continued)

Symptom	Possible Cause	Solution
Active and standby Cisco VSGs are not synchronized.	**Incompatible versions** The boot variables for active and standby Cisco VSGs are set to different image names, or if image names are the same, the files are not the correct files. When active and standby Cisco VSGs are running different versions that are not HA compatible, they are unable to synchronize.	Update the software version or the boot variables as follows: 1. From each Cisco VSG (active and standby), verify the software version by entering the **show version** command. 2. Reload the standby Cisco VSG with the version that is running the active Cisco VSG by doing one of the following: – Correct the boot variable names. – Replace the incorrect software files. See the "Reloading a Module" section on page 7-8.
	Broadcast traffic problem The broadcast traffic from the standby to the active Cisco VSG might prevent the Cisco VSGs from synchronizing. The standby Cisco VSG tries to contact the active Cisco VSG periodically, but if broadcast traffic problems persist for over a minute when the standby is booting up, the system cannot synchronize.	Fix the traffic problem and reload the standby Cisco VSG as follows. 1. From the standby Cisco VSG, verify the broadcast traffic problem by entering the **show system internal log sysmgr verctrl** command. The following message appears: `standby_verctrl: no response from the active System Manager` 2. Fix network connectivity. 3. Reload the standby Cisco VSG by using the **reload module** *standby_module_number* command. See the "Reloading a Module" section on page 7-8.

	False standby removal
	The active Cisco VSG falsely detects a disconnect with the standby. The standby is removed and reinserted and synchronization does not occur.
		Verify redundancy states and reload the standby Cisco VSG as follows:
		1. Verify active Cisco VSG redundancy by using the **show system internal redundancy status** command. The output is as follows:
		`RDN_DRV_ST_AC_NP`
		2. Verify the standby Cisco VSG redundancy by using the **show system internal redundancy status** command. The output is as follows:
		`RDN_DRV_ST_SB_AC`
		3. Reload the standby Cisco VSG by using the **reload module** *standby_module_number* command.
		See the "Reloading a Module" section on page 7-8.

Table 7-1 Problems with High Availability (continued)

Symptom	Possible Cause	Solution
The Cisco VSG HA pair reboots continuously in headless mode (VSMs are down).	The nonsystem VLAN Cisco VSG ports are down after they reconnect post reboot of the Cisco VSG because the VSM is not present to bring them up.	Check if the service and HA VLANs are configured as system VLANs. If they are not system VLANs and the Cisco VSG pair reboots for any reason, they do not come back up until the VSM comes up.

High Availability Troubleshooting Commands

This section lists commands that you can use to troubleshoot problems related to high availability.

This section includes the following topics:

Checking the Domain ID of the Cisco VSG

You can display the domain ID information by entering the **show vsg** command.

This example shows how to display the domain ID information:

```
vsg# show vsg
Model: VSG
HA ID: 3000
VSG Software Version: 4.2(1)VSG1(1) build [4.2(1)VSG1(1)]
PNSC IP: 10.193.75.145
```

Checking Redundancy

This section includes the following topics:

Checking the System Redundancy Status

You can check the system redundancy status by entering the **show system redundancy status** command.

This example shows how to display the system redundancy status:

```
vsg# 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 VSG
    Supervisor state:   Active
      Internal state:   Active with HA standby

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

Checking the System Internal Redundancy Status

You can check the system internal redundancy status by entering the **show system internal redundancy info** command.

This example shows how to display the system internal redundancy status information:

```
vsg# show system internal redundancy info
My CP:
  slot: 0
  domain: 184 <-- Domain id used by this VSG
  role:   primary <-- Redundancy role of this VSG
  status: RDN_ST_AC <-- Indicates redundancy state (RDN_ST) of the this VSG 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 VSG 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 VSG 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 VSG 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
```

Checking the System Manager State

You can check the system internal sysmgr state by entering the **show system internal sysmgr state** command.

This example shows how to display the system internal sysmgr state information:

```
vsg# 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
```

Reloading a Module

You can reload a module by entering the **reload module** command.

Note Using the **reload** command without specifying a module reloads the whole system.

This example shows how to reload a module:

```
vsg# reload module 2
This command will reboot the system (y/n)? y
```

Attaching to the Standby Cisco VSG Console

The standby Cisco VSG console is not accessible externally. You can access the standby Cisco VSG console through the active Cisco VSG by entering the **attach module** *module-number* command.

This example shows how to access the standby Cisco VSG through the active Cisco VSG:

```
vsg# attach module 2
Attaching to module 2...
To exit type 'exit', to abort type '$.'
Cisco Nexus Operating System (NX-OS) Software
TAC support: http://www.cisco.com/tac
Copyright (c) 2002-2011, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained in this software are
owned by other third parties and used and distributed under
license. Certain components of this software are licensed under
the GNU General Public License (GPL) version 2.0 or the GNU
Lesser General Public License (LGPL) Version 2.1. A copy of each
such license is available at
http://www.opensource.org/licenses/gpl-2.0.php and
http://www.opensource.org/licenses/lgpl-2.1.php
```

Checking for the Event History Errors

You can check for errors in the event history by entering the **show system internal sysmgr event-history errors** command.

This example shows how to display errors that have been logged in the event history:

```
vsg# show system internal sysmgr event-history errors
Event:E_DEBUG, length:122, at 370850 usecs after Thu Feb  3 09:45:28 2011
[101] sysmgr_sdb_set_standby_state: Setting standby super state in sdb for vdc 1 to
SYSMGR_SUPERSTATE_STABLE, returned
0x0

Event:E_DEBUG, length:73, at 408277 usecs after Thu Feb  3 09:44:52 2011
[101] active_gsyncctrl_info_parse: UUID 0xB6 in vdc 1 service not running

Event:E_DEBUG, length:73, at 593428 usecs after Thu Feb  3 09:44:49 2011
[101] active_gsyncctrl_info_parse: UUID 0xE0 in vdc 1 service not running

Event:E_DEBUG, length:80, at 624613 usecs after Thu Feb  3 09:44:40 2011
[101] process_plugin_load_complete_msg: Start done rcvd for all plugins in vdc 1

Event:E_DEBUG, length:89, at 624611 usecs after Thu Feb  3 09:44:40 2011
[101] process_plugin_load_complete_msg: Received plugin start done for plugin 1 for vdc 1

Event:E_DEBUG, length:99, at 518152 usecs after Thu Feb  3 09:44:39 2011
[101] perform_bootup_plugin_manager_interactions: all bootup plugins have now been loaded
in vdc: 1

Event:E_DEBUG, length:79, at 518150 usecs after Thu Feb  3 09:44:39 2011
[101] perform_bootup_plugin_manager_interactions:incrementing number of plugins

Event:E_DEBUG, length:118, at 518020 usecs after Thu Feb  3 09:44:39 2011
[101] perform_bootup_plugin_manager_interactions: plugin has been loaded in vdc 1 -
sending response to Plugin Manager

Event:E_DEBUG, length:58, at 631599 usecs after Thu Feb  3 09:44:38 2011
[101] process_reparse_request: on vdc 1, plugin start rcvd
```

Standby Synchronization

This section includes the following topic:

Synchronization Fails

If the standby Cisco VSG is stuck in the synchronization stage, follow these steps on the active Cisco VSG:

Step 1 Enter the **show system internal sysmgr state** command and check for a line similar to the following:

```
Gsync in progress for uuid: xxxx
```

If this entry is present and shows the same xxxx value for a long time, the system has trouble synchronizing the state for one of the processes.

Step 2 Identify the process by entering the **show system internal sysmgr service running | grep xxxx** command.

This message appears in the first few lines of the output:

```
BL-bash# show system internal sysmgr state
The master System Manager has PID 1350 and UUID 0x1.
Last time System Manager was gracefully shutdown.
Gsync in progress for uuid: 0x18
The state is SRV_STATE_MASTER_ACTIVE_HOTSTDBY entered at time Mon Feb 21 17:56:3
9 2011.

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

If the synchronization for each process occurs quickly, you might not have the chance to see the line (you might have to enter the command repeatedly as the standby Cisco VSG). If gsync for a particular process gets stuck, the line stays in the output for a while.

Step 3 If you are able to login to the console of the standby Cisco VSG (you might need to press **Ctrl-C** after giving the password), check the output of these two commands:

- **show system internal sysmgr state**
- **show system internal sysmgr service running | grep xxxx**
  where xxxx is from the line “Gsync in progress for uuid: xxxx” (found by using the **show system internal sysmgr state** command)

Step 4 If access to the system is available only after the standby server has booted up and synchronized with the active server, use the following commands:

- Enter the **show system internal sysmgr bootupstats** command and look for processes that took much longer than the rest, in the time that the system took to boot.
- On the standby console, enter the **show system internal sysmgr gsyncstats** command and look for processes with large Gsync time values.

Troubleshooting System Issues

This chapter describes how to troubleshoot Cisco Virtual Security Gateway (VSG) system issues.

This chapter includes the following sections:

Information About the System

The Cisco VSG provides firewall functionality for the VMs that have the vEths with port profiles created by the Virtual Supervisor Module (VSM). To allow the Cisco VSG to function properly, the Cisco VSG should have registered with a Cisco Prime Network Services Controller (PNSC) and the Cisco VSG data interface MAC address should be seen by the VSM.

The example shows how to display information about the system:

```
vsg# show vsg
Model: VSG
HA ID: 218
VSG Software Version: 4.2(1)VSG1(1) build [4.2(1)VSG1(1)]
PNSC IP: 10.193.77.223
VSG-PERF-1_1#
VSG-PERF-1_1# show nsc-pa status
NSC Policy-Agent status is - Installed Successfully. Version 1.0(1j)-vsg
```

Make sure that the Cisco VSG MAC address is learned by the VSM by entering the **show vservice node detail** command as follows:

```
vsm# show vservice node detail
#Node Information
#Node ID:1      Name:vasatDbd5
 Type:asa       IPAddr:172.8.8.201      Fail:open   Vxlan:bd5555
 Mod  State     MAC-Addr                VVer
   4  Alive     00:50:56:b5:37:8f          2
#Node ID:13     Name:vsg12tD104
 Type:vsg       IPAddr:10.10.10.104     Fail:open   Vlan:504
 Mod  State     MAC-Addr                VVer
   4  Alive     00:50:56:b5:6d:36          2
```

```
6  Alive     00:50:56:b5:6d:36          2
```

For more information, see the following documents for your release number:

- *Cisco Virtual Security Gateway*
- *Cisco Virtual Network Management Center*
- *Quick Start Guide for Cisco Virtual Security Gateway and Virtual Network Management Center*

Problems with VM Traffic

When troubleshooting problems with intrahost 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 addresses as that are 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 one uplink sharing a VLAN with the VMware virtual NIC. If there is more than one uplink, they must be in a port channel.
- Ping an SVI on the upstream switch by entering the **show intX counters** command.

VEM Troubleshooting Commands

This section includes the following topics:

Displaying VEM Information

Use the following commands to display Virtual Ethernet Module (VEM) information:

- **vemlog**—Displays and controls VEM kernel logs
- **vemcmd**—Displays configuration and status information
- **vem-support all**—Displays support information
- **vem status**—Displays status information
- **vem version**—Displays version information
- **vemcmd show arp all**—Displays the ARP table on the VEM
- **vemcmd show vsn config**—Displays all the Cisco VSGs configured on the VEM and the Cisco VSG licensing status (firewall on or off)
- **vemcmd show vsn binding**—Displays all of the VM LTL ports to the Cisco VSG bindings
- **vemcmd show learnt**—Displays all of the VMs that have been learned by the VEM

Displaying Miscellaneous VEM Details

These commands provide additional VEM details:

- **vemlog show last** *number-of-entries*—Displays the circular buffer

  This example shows how to display the number of entries in the circular buffer:

```
[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

  This example shows how to display log entries:

```
[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

  This example shows how to display the vemcmd help:

```
[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**—Clears 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

You can display the list of debug filters by entering the **vemlog show debug | grp vpath** command.

This example shows how to display the list of debug filters:

```
~ # vemlog show debug | grep vpath
              vpath       ENWID P ( 95)       ENW     (  7)
```

```
    vpathapi       ENWID P ( 95)       ENW     (  7)
     vpathfm       ENWID P ( 95)       ENW     (  7)
    vpathfsm       ENWID P ( 95)       ENW     (  7)
  vpathutils       ENWID P ( 95)       ENW     (  7)
    vpathtun       ENWID P ( 95)       ENW     (  7)
```

Troubleshooting the Cisco VSG in the Layer 3 Mode

This section includes the following topics:

show vservice node brief Command Output Indicates Service Node State is Down

This section includes the following topics:

Cisco VSG with a VN Service vmknic in Layer 3 Mode

When encapsulated traffic that is destined to a Cisco VSG is connected to a different subnet other than the vmknic subnet, the VEM does not use the VMware host routing table. Instead, the vmknic initiates an Address Resolution Protocol (ARP) for the remote Cisco VSG IP addresses.

You must configure the upstream router to respond by using the proxy ARP feature. If the proxy ARP feature is not configured on the upstream router, the ARP fails and the **show vservice node brief** indicates that the service node state is down.

To resolve this issue configure the proxy ARP feature on the router as follows:

```
sg-cat3k-L14-qa(config)# int vlan 3756
sg-cat3k-L14-qa(config-if)# ip proxy-arp
sg-cat3k-L14-qa(config-if)# end
sg-cat3k-L14-qa# sh ip int vlan 3756 | inc Proxy
  Proxy ARP is enabled
  Local Proxy ARP is disabled
sg-cat3k-L14-qa#
```

Cisco VSGs with Multiple l3-vn-service vmknics in Layer 3 Mode

The data path traffic and the ARP packets for the Cisco VSGs in Layer 3 mode can use any vmknic that is configured on the VEM host for packet forwarding to the Cisco VSG when you enter the **capability l3-vservice** command.

Therefore, all vmknics that are on a VEM host must be able to reach all Cisco VSGs in Layer 3 mode.

If a router is between the vmknics and the Cisco VSGs, all vmknics must have an interface in the router network (VLAN), and all the Cisco VSGs in the Layer 3 mode must have an interface in the router network (VLAN) to ensure that each vmknic has a route to each Cisco VSG.

To resolve this issue ensure that all l3-vn-service vmknics can reach all the Cisco VSGs in the Layer 3 mode that are used by the VEM host.

Note You must enable Proxy ARP on all the interfaces of the router that is alongside the vmknics.

Traffic with Large Payloads Fails: ICMP Too Big Message Does Not Reach the Client with the Cisco VSG in Layer 3 Mode

If a router lies between the vmknic and the Cisco VSG in the Layer 3 mode, and the router receives a packet that it cannot forward due to a large packet size, the router generates an ICMP Too Big message for the vmknic. The vmknic cannot forward the ICMP Too Big message of the router to the client and the vmknic drops the message. The client never receives the ICMP Too Big message and cannot refragment the packet for successful end-to-end traffic and the end-to-end traffic fails. This problem is typically seen if the router interface to the VEM is set at a higher maximum transmission unit (MTU) than the router interface to the Cisco VSG. For example, the router interface to the VEM has an MTU of 1600 and the interface to the Cisco VSG has an MTU of 1500.

This problem can be seen as an increase in the ICMP Too Big Rcvd counter in the **show vservice statistics** command.

To resolve this issue, configure an oversized MTU (for example, 1600) on both of the router interfaces.

If L3 pre-frag is enabled, traffic is allowed and fragmentation happens, even when MTU is less than 1582. If L3 pre-frag is disabled, traffic fails for MTU less than 1582.

End-to-End Traffic with the Cisco VSG in Layer 3 Mode and Jumbo Frames Fails

If L3 Pre-frag is enabled, traffic is allowed. If L3 Pre-frag is disabled, traffic fails.

If jumbo frames are enabled in the network and the end-to-end traffic fails, make sure that the MTU of the client and server VMs are 82 bytes smaller than the uplink. For example, if the uplink MTU is 9000, set the MTU of the client and server VMs to 8918 bytes.

TCP State Checks

By default, TCP state checks are enabled in vPath for the traffic protected by the Cisco VSG. Sometimes, you might see delays in the TCP traffic. You can disable TCP state checks to diagnose the issue.

Check the following counters at the VSM in the **show vservice statistics** output:

```
vsm# show vservice statistics | grep "TCP chkfail"
```

```
  TCP chkfail InvalACK   0          TCP chkfail SeqPstWnd   0
  TCP chkfail WndVari        0
```

This example shows how to disable the TCP state checks on a VSM:

```
VSM(config)# vservice global type vsg
VSM(config-vsn)# no tcp state-checks
VSM(config-vsn)#
```

Connection Limit in the Cisco VSG

The Cisco VSG can have up to 256,000 active connections at any given point of time. If for some reason new connections slows down or connections see too many failures, you can check the Cisco VSG for any connection limits that it experiences. If the VEM-to-Cisco VSG connection is not smooth or have some issues that indicates that the Cisco VSG might have missed a few updates from vPath which results in an accumulation of large active connections in its flow table.

This example shows how to check the active connection count on the Cisco VSG:

```
vsg# show service-path statistics | inc "Active Connections"
Active Flows                       48  Active Connections               24
```

Debugging the Traffic Flow Via a Service Chain

When configured, the service-chain functionality enables traffic to flow through the Cisco VSG and the Cisco ASA 1000V cloud firewall. The Cisco VSG monitors the data packets and authorizes its flow from the VM to the destination ports. The VM and Cisco ASA 1000V are always in the same broadcast domain, that is, either a VLAN or a Virtual Extensible Local Area Network (VXLAN).

To debug the traffic flow via the service chain, follow these steps:

Step 1 Make sure that the VM's default gateway is set to the ASA 1000V inside interface and is reachable.

Step 2 On the VSM, ensure that the Cisco VSG and ASA 1000V are alive, which ensures that the vPath is able to reach the service nodes.

```
vsm# show vservice node brief
#Node Information
 ID Name Type   IP-Address      Mode   State   Module
  2  VSG vsg    192.168.10.1    v-140  Alive   3,4
  6 ASA asa    3.3.3.1 v-200  Alive   3,4
```

Step 3 On the VSM, check a connection's status of action (SAct).

```
vsm# show vservice connection
# Module 1
Proto SrcIP[:Port] SAct  DstIP[:Port]    DAct  Flags Bytes
icmp 192.168.10.15 Pp     192.168.11.15             882
```

In the SAct value Pp, the uppercase 'P' indicates the action that is initiated by the Cisco VSG, while the lowercase 'p' indicates the action that is deduced based on the returning traffic from the ASA V1000. If the SAct value is 'rr,' it indicates that the traffic is redirecting to either the Cisco VSG or the ASA V1000 but no response is being received.

Step 4 On the VSM, verify that the service node version information (VVer) is '2' so that it works in the service-chain.

```
vsm# show vservice node detail
```

```
#Node Information
#Node ID:2      Name:VSG
 Type:asa       IPAddr:192.168.10.1     Fail:open   Vlan:140
 Mod  State     MAC-Addr                VVer
   3  Alive     00:50:56:a6:02:a5          2
   4  Alive     00:50:56:a6:02:a5          2
#Node ID:6      Name:ASA
 Type:asa       IPAddr:3.3.3.1          Fail:open   Vlan:200
 Mod  State     MAC-Addr                VVer
   3  Alive     00:50:56:a6:02:6d          2
   4  Alive     00:50:56:a6:02:6d          2
```

Troubleshooting the Service Chain by Excluding the Cisco VSG Node

The service-chain configuration has the Cisco VSG and ASA 1000V nodes in its service path for a given traffic flow. For debugging purposes, the Cisco VSG can be removed temporarily from the node configuration to isolate a problem. Thus, a user can verify the traffic flow with just the ASA 1000V. Later, the Cisco VSG can be added again to restore the original service-chain configuration using the two said service nodes.

VEM/vpath Configured Correctly on a VEthernet Interface for a ServiceChain

You can use the **module vem** *vem-num* **execute vemcmd show vservice bindings** command on the VSM to ensure that the bindings are correctly configured on the VEM for a service chain. Two entries appear for a single LTL—one of each service node must be displayed.

```
vsm# module vem 3 execute vemcmd show vsn bindings
  VSG Services Enabled | VSG Licenses Available   2
  ASA Services Enabled | ASA Licenses Available   2
    LTL  PATH   VSN  SWBD     IP          P-TYPE  P-ID
    60     6     2  3756     10.10.10.202       1 49
    60     6    33  3770      172.31.2.11       2 52 >> two service node bindings for LTL
60 and 62
    62     6     2  3756     10.10.10.202       1 49
    62     6    33  3770      172.31.2.11       2 52
```

Cisco VSG on a VXLAN is not working

The Cisco VSG node can be configured with a VXLAN in the Layer 2 mode only. Make sure that the adjacency is correctly defined as Layer 2 and the bridge-domain configuration is valid. The **show service node brief** command can be used to check a service node's state with respect to the vPath.

This example shows a Cisco VSG node configuration for a VXLAN:

```
vservice node VSG-vxlan-33071 type vsg
  ip address 20.20.20.182
  adjacency l2 vxlan bridge-domain 33071
  fail-mode close
```

This example shows how to display the node status in a VXLAN:

```
vsm# show vservice node brief
#Node Information
 ID Name           Type   IP-Address      Mode   State   Module
```

```
35  VSG-vxlan-33071vsg    20.20.20.182    vxlan  Alive   3,4,5
```

Troubleshooting Cisco VSG Flow Issues on KVM VEM Module

This chapter describes how to troubleshoot Cisco Virtual Security Gateway (VSG) flow issues on KVM VEM module.

This chapter includes the following sections:

Understanding KLM Flow Messages

The Cisco vPath support on KVM is limited to a VSG type service node. The flows are offloaded to the KLM when the VSG decides to offload a PERMIT or DENY action to the VEM. When offloaded, KLM flows with following actions are created: vpath_permit, vpath_permit_tcp, and vpath_deny. Table 9-1 lists the messages generated:

Table 9-1 KLM Flow Messages

KLM Flow	Information
ICMP deny flow	key=in_port:21,vlan:120,dmac:06:0d:eb:00:80:01,smac:06:0d:eb:00:70:01,etype:0x0800,dip:172.23.128.8,sip:172.23.128.7,proto:1,tos:0,dport:0,sport:8 **actions=vpath_deny:** pkts=1 bytes=98 drops=1 punts=0
ICMP permit flow	key=in_port:21,vlan:120,dmac:06:0d:eb:00:80:01,smac:06:0d:eb:00:50:01,etype:0x0800,dip:172.23.128.8,sip:172.23.128.5,proto:1,tos:0,dport:0,sport:8 **actions=vpath_permit**: pkts=10 bytes=980 drops=0 punts=0
UDP permit flow	key=in_port:51,vlan:120,dmac:06:0d:eb:00:50:01,smac:06:0d:eb:00:80:01,etype:0x0800,dip:172.23.128.5,sip:172.23.128.8,proto:17,tos:0,dport:47161,sport:44260 **actions=vpath_permit**: pkts=1003114 bytes=1452509072 drops=0 punts=0
TCP permit flow	key=in_port:21,vlan:120,dmac:06:0d:eb:00:80:01,smac:06:0d:eb:00:50:01,etype:0x0800,dip:172.23.128.8,sip:172.23.128.5,proto:6,tos:0,dport:2083,sport:59759 **actions=vpath_permit_tcp**:0141000001000000 pkts=4 bytes=292 drops=0 punts=0

Troubleshooting TCP State Connection Objects

When TCP permit flows are offloaded to the KLM, connection objects are programmed in the KLM to facilitate TCP state verification, which is performed as part of the vpath_permit_tcp action.You can use the **vem cmd show klm vpath** command to list statistics related to TCP state connection objects:

```
[root@kvm-cuda5 ~]# vemcmd show klm vpath
num_conns: 2
num_conn_adds: 27
num_conn_dels: 25
num_conn_gets: 152
num_conn_sets: 152
```

where,

num_conns: Indicates the number of connection objects currently programmed in the KLM.

Note The remaining stats indicate the number of times operations have been performed to add, delete, fetch, and set connection objects in the KLM.

Before Contacting Technical Support

This chapter describes the steps to take before calling for technical support.

This chapter includes the following sections:

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/en/US/support/tsd_cisco_worldwide_contacts.html

Gathering Information for Technical Support

Use this procedure to gather information about your network that you will provide to your customer support representative or Cisco TAC.

Note Required logs and counters are part of volatile storage and do not persist through a reload. Do not reload the module or the switch until you have completed this procedure.

DETAILED STEPS

Step 1 Configure your Telnet or Secure Shell (SSH) application to log screen output to a text file.

Step 2 Set the number of lines that appear on the screen so that pausing is disabled:

terminal length 0

Step 3 Display the configuration information needed to troubleshoot your network by entering the **show tech-support** command.

Step 4 Capture the error codes that appear in your message logs by entering the following commands:

- **show logging logfile**—Displays the contents of the logfile.
- **show logging last** *number*—Displays the last few lines of the logfile.

Step 5 Gather your answers to the following questions:

- On which Cisco VSG is the problem occurring?
- Are Cisco Virtual Security Gateway (VSG) software, driver versions, operating systems versions, and storage device firmware in your fabric?
- Are you running ESX and vCenter Server software?
- What is your network topology?
- Did you make any changes to the environment (VLANs, adding modules or upgrades) before 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 tried? Which, if any, of the following tools were used?
  - Ethanalyzer, local or remote SPAN
  - CLI debug commands
  - traceroute, ping
- Is your problem related to a software upgrade attempt?
  - What was the original Cisco VSG version?
  - What is the new Cisco VSG version?

Obtaining a File of Core Memory Information

Cisco customer support engineers often use files from your system for analysis. One such file that contains memory information 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 TAC representative, and send it to a TFTP server so that it can be e-mailed to TAC.

This example shows how to generate a file of core memory information or a core dump:

```
vsg(config)# system cores tftp://10.91.51.200/svr15svc_cores
vsg(config)# show system cores
Cores are transferred to tftp://10.91.51.200/svr15svc_cores
```

Note The filename (indicated by svr15svc_cores) must exist in the TFTP server directory.

Copying Files

You might need to move files to or from the switch. These files may include log, configuration, or firmware files.

The Cisco VSG always acts as a client. For example, an FTP/SCP/TFTP session always originates from the switch and either pushes files to an external system or pulls 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 4 transfer protocols and 12 different sources for files.

This example shows the copy options:

```
vsg# copy ?
    bootflash:      Select source filesystem
    core:           Select source filesystem
    debug:          Select source filesystem
    ftp:            Select source filesystem
    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
    startup-config  Copy startup configuration to destination
    system:         Select source filesystem
    tftp:           Select source filesystem
    volatile:       Select source filesystem
```

This example shows how to use secure copy (SCP) as the transfer mechanism:

```
vsg# scp:[//[username@]server][/path]
```

This example shows how to copy /etc/hosts from 203.0.113.11 using the user user1, where the destination is hosts.txt:

```
vsg# copy scp://user1@203.0.113.11/etc/hosts bootflash:hosts.txt
user1@203.0.113.11's password:
hosts 100% |***************************| 2035 00:00
```

This example shows how to back up the startup configuration to an SFTP server:

```
vsg# copy startup-config sftp://user1@203.0.113.11/test/startup-configuration.bak1
Connecting to 203.0.113.11...
User1@203.0.113.11's password:
```

Tip You should back up the startup-configuration file to a server daily and before you make any changes. You could use a short script to be run on the Cisco VSG to perform a save and a backup of the configuration. The script must contain two commands: **copy running-configuration startup-configuration** and **copy startup-configuration tftp://***server/name*. To execute the script, use the **run-script** [*filename*] command.