Cisco Nexus 1000V System Management Configuration Guide, Release 4.0(4)SV1(3)

First Published: 2011-04-27

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.

New and Changed Information

This chapter lists new and changed content by release, and where it is located in this document.

Table 1   New and Changed Information for Release 4.0(4)SV1(3)

Content	Description	Changed in Release	Where Documented
Network Analysis Module (NAM)	NAM support for NetFlow data sources	4.0(4)SV1(3)	"Configuring NetFlow"
	NAM support for ERSPAN data sources	4.0(4)SV1(3)	"Configuring Local SPAN and ERSPAN"
ERSPAN Type-III header	The ERSPAN Type-III extended format header frame enhances support for network management, intrusion detection, and lawful intercept.	4.0(4)SV1(3)	"Configuring Local SPAN and ERSPAN"

Table 2   New and Changed Information for Release 4.0(4)SV1(2)

Content	Description	Changed in Release	Where Documented
Layer 3 Control	Allows a VSM to be Layer 3 accessible and control hosts that reside in a separate Layer 2 network.	4.0(4)SV1(2)	"Configuring the Domain"
iSCSI Multipath	Allows multiple routes between a server and its storage devices.	4.0(4)SV1(2)	"Configuring iSCSI Multipath"
Recommended Reading	Lists reading recommended before configuring the Cisco Nexus 1000V.	4.0(4)SV1(2)	"Preface"
Configuration Limits	Lists the configuration limits for the Cisco Nexus 1000V.	4.0(4)SV1(2)	"Configuration Limits"

System Management Overview

CDP

Cisco Discovery Protocol (CDP) runs over the data link layer and is used to advertise information to all attached Cisco devices, and to discover and view information about attached Cisco devices. CDP runs on all Cisco-manufactured equipment.

For more information about CDP, see Chapter 2, "Configuring CDP."

Domains

You must create a domain name for Cisco Nexus 1000V and then add control and packet VLANs for communication and management. This process is part of the initial setup of the a Cisco Nexus 1000V when installing the software. If you need to create a domain later, you can do so using the **setup** command or the procedures in Chapter 3, "Configuring the Domain."

You can establish Layer 3 Control in your VSM domain so that your VSM is Layer 3 accessible and able to control hosts that reside in a separate Layer 2 network. For more information, see the "About Layer 3 Control" section on page 3-1.

Server Connections

In order to connect to vCenter Server or an ESX server, you must first define the connection in the Cisco Nexus 1000V. Chapter 4, "Managing Server Connections" describes how to connect and disconnect with VCenter Server and viewing connections.

Configuration Management

The Cisco Nexus 1000V lets you change the switch name, configure messages of the day, and display, save, and erase configuration files. For more information about managing the configuration, see Chapter 5, "Managing the Configuration."

File Management

Using a single interface, you can manage the file system including:

- Flash memory file systems
- Network file systems (TFTP and FTP)
- Any other endpoint for reading or writing data (such as the running configuration)

For more information about working with files, see Chapter 6, "Working with Files."

User Management

You can identify the users currently connected to the device and send a message to either a single user or all users. For more information, see Chapter 7, "Managing Users."

NTP

The Network Time Protocol (NTP) synchronizes timekeeping among a set of distributed time servers and clients. This synchronization allows you to correlate events when you receive system logs and other time-specific events from multiple network devices.

For more information about NTP, see Chapter 8, "Configuring NTP."

Local SPAN and ERSPAN

The Ethernet switched port analyzer (SPAN) lets you monitor traffic in and out of your device, and duplicate packets from source ports to destination ports.

For information about configuring SPAN, see Chapter 9, "Configuring Local SPAN and ERSPAN."

You can also use the Cisco Network Analysis Module (NAM) to monitor ERSPAN data sources for application performance, traffic analysis, and packet header analysis.

To use NAM for monitoring the Cisco Nexus 1000V ERSPAN data sources see the *Cisco Nexus 1010 Network Analysis Module Installation and Configuration Note, 4.2*.

SNMP

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

For more information about SNMP, see Chapter 10, "Configuring SNMP."

NetFlow

NetFlow gives visibility into traffic transiting the virtual switch by characterizing IP traffic based on its source, destination, timing, and application information. This information is used to assess network availability and performance, assist in meeting regulatory requirements (compliance), and help with troubleshooting.

For more information, see Chapter 11, "Configuring NetFlow."

You can also use the Cisco Network Analysis Module (NAM) to monitor NetFlow data sources. For more information see the *Cisco Nexus 1010 Network Analysis Module Installation and Configuration Note, 4.2*.

System Messages

You can use system message logging to control the destination and to filter the severity level of messages that system processes generate. You can configure logging to a terminal session, a log file, and syslog servers on remote systems.

System message logging is based on RFC 3164. For more information about the system message format and the messages that the device generates, see the *Cisco NX-OS System Messages Reference*.

For information about configuring system messages, see Chapter 12, "Configuring System Message Logging."

iSCSI Multipath

The iSCSI multipath feature sets up multiple routes between a server and its storage devices for maintaining a constant connection and balancing the traffic load.

For more information, see Configuring iSCSI Multipath, page 13-1.

Troubleshooting

Ping and traceroute are among the available troubleshooting tools.

For more information, see the *Cisco Nexus 1000V Troubleshooting Guide, Release 4.0(4)SV1(3)*.

Configuring CDP

This chapter describes how to configure the Cisco Discovery Protocol (CDP).

Information About CDP

Cisco Discovery Protocol (CDP) runs over the data link layer and is used to advertise information to all attached Cisco devices, and to discover and view information about attached Cisco devices. CDP runs on all Cisco-manufactured equipment.

CDP gathers protocol addresses of neighboring devices and discovers the platform of those devices. CDP runs over the data link layer only. Two systems that support different Layer 3 protocols can learn about each other.

Each device you configure for CDP sends periodic advertisements to a multicast address. Each device advertises at least one address at which it can receive SNMP messages. The advertisements also contain hold-time information, which indicates the length of time that a receiving device should hold CDP information before discarding it. You can configure the advertisement or refresh timer and the hold timer.

CDP Version 2 (CDPv2) allows you to track instances where the native VLAN ID or port duplex states do not match between connecting devices.

CDP advertises the following type-length-value fields (TLVs):

- Device ID
- Address
- Port ID
- Capabilities
- Version
- Platform
- Native VLAN
- Full/Half Duplex
- MTU
- SysName
- SysObjectID
- Management Address
- Physical Location

All CDP packets include a VLAN ID. The CDP packet is untagged, so it goes over the native/access VLAN, which is then also added to the packet.

For more information on VLANs, see the *Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.0(4)SV1(3).*

High Availability

Stateless restarts are supported for CDP. After a reboot or a supervisor switchover, the running configuration is applied.

Guidelines and Limitations

CDP has the following configuration guidelines and limitations:

- CDP can discover up to 256 neighbors per port if the port is connected to a hub with 256 connections.

- CDP must be enabled globally before you can configure CDP on an interface. CDP is enabled globally by default, but can be disabled using the "Enabling or Disabling CDP Globally" procedure on page 2-3.

- You can configure CDP on physical interfaces and port channels only.

Defaults

Table 2-1 lists the CDP default settings.

Table 2-1   CDP Defaults

Parameters	Default
CDP	Enabled globally and on all interfaces
CDP version	Version 2
CDP device ID	System name
CDP timer	60 seconds
CDP hold timer	180 seconds

Configuring CDP

This section includes the following topics:

- CDP Global Configuration, page 2-3

- Enabling CDP on an Interface, page 2-7

- Disabling CDP on an Interface, page 2-8

CDP Global Configuration

This section includes the following topics:

Enabling or Disabling CDP Globally

Use this procedure to enable or disable CDP globally.Although CDP is enabled globally by default, should it be disabled, you can use this procedure to enable it again.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in EXEC mode.
- CDP must be enabled globally before you can configure it on an interface.
- When you globally disable the CDP feature, all CDP configurations are removed.

SUMMARY STEPS

1. **config t**
2. **[no] cdp enable**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	`[no] cdp enable`
	
	Example:
	`n1000v(config)# cdp enable`
	
	Example:
	`n1000v(config)# no cdp enable`
		Enables or disables the CDP feature globally.

Advertising a CDP Version

Use this procedure to designate the CDP version to advertise on the device.

BEFORE YOU BEGIN

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

- You know the version of CDP currently supported on the device.

- Only one version of CDP (version 1 or version 2) is advertised at a time for all uplinks and port channels on the switch.

- For more information about CDP, see the "Information About CDP" section on page 2-1.

SUMMARY STEPS

1. **config t**

2. **cdp advertise {v1 | v2}**

3. (Optional) **show cdp global**

4. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	`Example:`
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	`cdp advertise {v1	v2}`  `Example 1:` `n1000v(config)# cdp advertise v1` `n1000v(config)#`  `Example 2:` `n1000v(config)# cdp advertise v2` `n1000v(config)#`	Assigns the CPD version to advertise. • CDP Version 1 • CDP Version 2
Step 3	`show cdp global`	(Optional) Displays the CDP configuration, indicating the CDP version that is being advertised or sent to other devices.

Command	Purpose

Example 1:
`n1000v(config)# `**`show cdp global`**
`Global CDP information:`
`    CDP enabled globally`
`    Sending CDP packets every 60 seconds`
`    Sending a holdtime value of 180 seconds`
`    Sending CDPv2 advertisements is disabled`
`    Sending DeviceID TLV in Default Format`
 	
Example 2:
`n1000v(config)# `**`show cdp global`**
`Global CDP information:`
`    CDP enabled globally`
`    Sending CDP packets every 60 seconds`
`    Sending a holdtime value of 180 seconds`
`    Sending CDPv2 advertisements is enabled`
`    Sending DeviceID TLV in Default Format`	

Step 4 `copy running-config startup-config`
 	
Example:
`n1000v(config)# copy running-config`
`startup-config`
	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Configuring CDP Options

Use this procedure to configure the following for CDP:

- the device ID format to use

  ✎

  Note   Only the **system-name** device ID format is supported.

- the maximum hold time for neighbor information
- the refresh time for sending advertisements

BEFORE YOU BEGIN

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

- You can view output from upstream cat6k switch using the **show cdp neighbor** command.
- If you are setting the holdtime, you know how long you want CDP to retain neighbor information.
- If you are setting the CDP timer, you know how often you want CDP to advertise.
- For more information about CDP, see the "Information About CDP" section on page 2-1.

SUMMARY STEPS

1. **config t**
2. (Optional) **cdp format device-id system-name**
3. **show cdp neighbors** from the upstream device
4. **show cdp neighbors** from your device
5. (Optional) **cdp timer** *seconds*

6. (Optional) **cdp holdtime** *seconds*

7. (Optional) **show cdp global**

8. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	`cdp format device-id system-name`
	Example:
	`n1000v(config)# cdp format device-id`
	`system-name`
	`n1000v(config)#`
		(Optional) Specifies that CDP uses the system name for the device ID format.

Step 3	`show cdp neighbors`	Displays your device from the upstream device.

Example:
```
swordfish-6k-2#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone

Device ID          Local Intrfce      Holdtme      Capability  Platform  Port ID

02000c000000       Gig 1/16           14           S       Soft Swit Eth 2/4
02000c000000       Gig 1/17           14           S       Soft Swit Eth 2/5
02000c000000       Gig 1/14           14           S       Soft Swit Eth 2/2
02000c000000       Gig 1/15           14           S       Soft Swit Eth 2/3
02000c000000       Gig 1/18           13           S       Soft Swit
```

	Command	Purpose
Step 4	`show cdp neighbors`	Displays the upstream device from your device,

```
n1000v(config)# show cdp neighbors
Capability Codes: R - Router, T - Trans-Bridge, B - Source-Route-Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  V - VoIP-Phone, D - Remotely-Managed-Device,
                  s - Supports-STP-Dispute

Device ID             Local Intrfce   Hldtme  Capability  Platform     Port ID

swordfish-6k-2        Eth2/2          169     R S I    WS-C6503-E    Gig1/14
swordfish-6k-2        Eth2/3          139     R S I    WS-C6503-E    Gig1/15
swordfish-6k-2        Eth2/4          135     R S I    WS-C6503-E    Gig1/16
swordfish-6k-2        Eth2/5          177     R S I    WS-C6503-E    Gig1/17
swordfish-6k-2        Eth2/6          141     R S I    WS-C6503-E    Gig1/18
```

	Command	Purpose
Step 5	`cdp holdtime` *seconds* **Example:** `n1000v(config)# cdp holdtime 10`	(Optional) Sets the maximum amount of time that CDP holds onto neighbor information before discarding it. • The range is from 10 to 255 seconds. • The default is 180 seconds.

	Command	Purpose
Step 6	`cdp timer` *seconds* **Example:** `n1000v(config)# cdp timer 5`	(Optional) Sets the refresh time for CDP to send advertisements to neighbors. • The range is from 5 to 254 seconds. • The default is 60 seconds.
Step 7	`show cdp global`	Displays the global CDP configuration.

	Example:
	`n1000v(config)# show cdp global`
	`Global CDP information:`
	`    CDP enabled globally`
	`    Sending CDP packets every 5 seconds`
	`    Sending a holdtime value of 10 seconds`
	`    Sending CDPv2 advertisements is disabled`
	`    Sending DeviceID TLV in Mac Address Format`	

Step 8	`copy running-config startup-config`
	
	Example:
	`n1000v(config-if)# copy running-config`
	`startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

CDP Interface Configuration

This section includes the following procedures:

Enabling CDP on an Interface

Use this procedure to enable CDP on a specific interface. Although CDP is enabled by default on all interfaces, should it become disabled, you can use this procedure to enable it again.

BEFORE YOU BEGIN

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

• The CDP feature is enabled globally. CDP is enabled globally by default, but can also be re-enabled using the "Enabling or Disabling CDP Globally" procedure on page 2-3.

• For more information about CDP, see the "Information About CDP" section on page 2-1.

SUMMARY STEPS

1. **config t**

2. **interface** *interface-type number*

3. **no cdp enable**

4. **cdp enable**

5. **show cdp interface** *interface-type number*

6. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**config t** **Example:** n1000v# config t n1000v(config)#	Places you in the CLI Global Configuration mode.
Step 2	**interface** *interface-type number* **Example:** n1000v(config)# interface port-channel 2 n1000v(config-if)#	Places you in the CLI Interface Configuration mode for the specific interface.
Step 3	**no cdp enable** **Example:** n1000v(config-if)# no cdp enable	Disables CDP on this interface.
Step 4	**cdp enable** **Example:** n1000v(config-if)# cdp enable	Enables CDP on this interface.
Step 5	**show cdp interface** *interface-type number* **Example:** n1000v(config-if)# show cdp interface mgmt0 mgmt0 is up CDP disabled on interface Sending CDP packets every 60 seconds Holdtime is 180 seconds	(Optional) Displays CDP information for the specified interface.
Step 6	**copy running-config startup-config** **Example:** n1000v(config-if)# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

This example shows how to enable CDP on port channel 2:

```
n1000v# config t
n1000v(config)# interface port-channel 2
n1000v(config-if)# no cdp enable
n1000v(config-if)# cdp enable
n1000v(config-if)# copy running-config startup-config
```

Disabling CDP on an Interface

Use this procedure to disable CDP on a specific interface.

BEFORE YOU BEGIN

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

- CDP is currently enabled on the device.

Note If CDP is disabled on the device, then it is also disabled for all interfaces.

- CDP is currently enabled on the specific interface you want to configure.

- For more information about CDP, see the "Information About CDP" section on page 2-1.

SUMMARY STEPS

1. **config t**

2. **interface** *interface-type* number

3. **no cdp enable**

4. (Optional) **show cdp interface** *interface-type* number

5. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	`interface` *interface-type number*
	
	Example:
	`n1000v(config)# interface mgmt0`
	`n1000v(config-if)#`
		Places you in the CLI Interface Configuration mode for the specified interface.

Step 3	`no cdp enable`
	
	Example:
	`n1000v(config-if)# no cdp enable`
		Disables CDP on the specified interface.

Step 4	`show cdp interface` *interface-type number* **Example:** `n1000v(config-if)# show cdp interface mgmt0`	(Optional) Displays CDP information for an interface.

Step 5	`copy running-config startup-config`
	
	Example:
	`n1000v(config-if)# copy running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

This example shows how to disable CDP on mgmt0:

```
n1000v# config t
n1000v(config)# interface mgmt0
n1000v(config-if)# no cdp enable
n1000v(config-if)# show cdp interface mgmt0
mgmt0 is up
    CDP disabled on interface
    Sending CDP packets every 60 seconds
    Holdtime is 180 seconds
n1000v(config-if)# copy running-config startup-config
```

Monitoring CDP

To monitor CDP traffic, use the following command:

Command	Purpose
show cdp traffic interface *interface-type slot/port*	Displays the CDP traffic statistics on an interface. See Example 2-7 on page 2-13

Clearing CDP Statistics

To clear CDP statistics, use one of the following commands.

Command	Purpose
clear cdp counters	Clears CDP statistics on all interfaces.
clear cdp counters interface *number*	Clears CDP statistics on the specified interface.
clear cdp table	Clears the CDP cache for one or all interfaces.

Verifying the CDP Configuration

To verify the CDP configuration, use one of the following commands:

Command	Purpose
show cdp all	Displays all interfaces that have CDP enabled. See Example 2-1 on page 2-10
show cdp entry {**all** | **name** *entry-name*}	Displays the CDP database entries. See Example 2-2 on page 2-11
show cdp global	Displays the CDP global parameters. See Example 2-4 on page 2-13
show cdp interface *interface-type slot/port*	Displays the CDP interface status. See Example 2-5 on page 2-13
show cdp neighbors {**detail** | **interface** *interface-type slot/port*}	Displays the CDP neighbor status. See Example 2-6 on page 2-13

Example 2-1   show cdp all

```
n1000v# show cdp all
Ethernet2/2 is up
    CDP enabled on interface
    Sending CDP packets every 60 seconds
    Holdtime is 180 seconds
Ethernet2/3 is up
    CDP enabled on interface
```

```
      Sending CDP packets every 60 seconds
      Holdtime is 180 seconds
Ethernet2/4 is up
      CDP enabled on interface
      Sending CDP packets every 60 seconds
      Holdtime is 180 seconds
Ethernet2/5 is up
      CDP enabled on interface
      Sending CDP packets every 60 seconds
      Holdtime is 180 seconds
Ethernet2/6 is up
      CDP enabled on interface
      Sending CDP packets every 60 seconds
      Holdtime is 180 seconds
mgmt0 is up
      CDP enabled on interface
      Sending CDP packets every 60 seconds
      Holdtime is 180 seconds
```

Example 2-2   show cdp entry *name*

```
n1000v# show cdp entry name swordfish-6k-2
----------------------------------------
Device ID:swordfish-6k-2
System Name:
Interface address(es):
    IPv4 Address: 172.28.30.2
Platform: cisco WS-C6503-E, Capabilities: Router Switch IGMP Filtering
Interface: Ethernet2/2, Port ID (outgoing port): GigabitEthernet1/14
Holdtime: 152 sec

Version:
Cisco IOS Software, s72033_rp Software (s72033_rp-IPBASE-M), Version 12.2(33)SXH2a,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Fri 25-Apr-08 09:11 by prod_rel_team
```

Example 2-3   show cdp entry all

```
n1000v# show cdp entry all
----------------------------------------
Device ID:swordfish-6k-2
System Name:
Interface address(es):
    IPv4 Address: 172.28.30.2
Platform: cisco WS-C6503-E, Capabilities: Router Switch IGMP Filtering
Interface: Ethernet2/2, Port ID (outgoing port): GigabitEthernet1/14
Holdtime: 140 sec

Version:
Cisco IOS Software, s72033_rp Software (s72033_rp-IPBASE-M), Version 12.2(33)SXH2a,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Fri 25-Apr-08 09:11 by prod_rel_team

Advertisement Version: 1

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

```
Device ID:swordfish-6k-2
System Name:
Interface address(es):
    IPv4 Address: 172.28.30.2
Platform: cisco WS-C6503-E, Capabilities: Router Switch IGMP Filtering
Interface: Ethernet2/3, Port ID (outgoing port): GigabitEthernet1/15
Holdtime: 129 sec

Version:
Cisco IOS Software, s72033_rp Software (s72033_rp-IPBASE-M), Version 12.2(33)SXH2a,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Fri 25-Apr-08 09:11 by prod_rel_team

Advertisement Version: 1


----------------------------------------
Device ID:swordfish-6k-2
System Name:
Interface address(es):
    IPv4 Address: 7.7.8.1
Platform: cisco WS-C6503-E, Capabilities: Router Switch IGMP Filtering
Interface: Ethernet2/4, Port ID (outgoing port): GigabitEthernet1/16
Holdtime: 154 sec

Version:
Cisco IOS Software, s72033_rp Software (s72033_rp-IPBASE-M), Version 12.2(33)SXH2a,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Fri 25-Apr-08 09:11 by prod_rel_team

Advertisement Version: 1


----------------------------------------
Device ID:swordfish-6k-2
System Name:
Interface address(es):
    IPv4 Address: 7.7.8.1
Platform: cisco WS-C6503-E, Capabilities: Router Switch IGMP Filtering
Interface: Ethernet2/5, Port ID (outgoing port): GigabitEthernet1/17
Holdtime: 156 sec

Version:
Cisco IOS Software, s72033_rp Software (s72033_rp-IPBASE-M), Version 12.2(33)SXH2a,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Fri 25-Apr-08 09:11 by prod_rel_team

Advertisement Version: 1


----------------------------------------
Device ID:swordfish-6k-2
System Name:
Interface address(es):
    IPv4 Address: 172.28.15.229
Platform: cisco WS-C6503-E, Capabilities: Router Switch IGMP Filtering
Interface: Ethernet2/6, Port ID (outgoing port): GigabitEthernet1/18
Holdtime: 171 sec

Version:
```

```
Cisco IOS Software, s72033_rp Software (s72033_rp-IPBASE-M), Version 12.2(33)SXH2a,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Fri 25-Apr-08 09:11 by prod_rel_team

Advertisement Version: 1
```

Example 2-4   show cdp global

```
n1000v(config)# show cdp global
Global CDP information:
    CDP enabled globally
    Sending CDP packets every 60 seconds
    Sending a holdtime value of 180 seconds
    Sending CDPv2 advertisements is disabled
    Sending DeviceID TLV in Default Format
```

Example 2-5   show cdp interface

```
n1000v(config)# show cdp interface ethernet 2/3
Ethernet2/3 is up
    CDP enabled on interface
    Sending CDP packets every 60 seconds
    Holdtime is 180 seconds
```

Example 2-6   show cdp neighbors interface

```
n1000v(config)# show cdp neighbors interface ethernet 2/3
Capability Codes: R - Router, T - Trans-Bridge, B - Source-Route-Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  V - VoIP-Phone, D - Remotely-Managed-Device,
                  s - Supports-STP-Dispute


Device ID           Local Intrfce   Hldtme  Capability  Platform      Port ID

swordfish-6k-2      Eth2/3          173     R S I       WS-C6503-E    Gig1/15
```

Example 2-7   show cdp traffic interface

```
n1000v(config)# show cdp traffic interface ethernet 2/3
----------------------------------------
Traffic statistics for Ethernet2/3
Input Statistics:
    Total Packets: 98
    Valid CDP Packets: 49
        CDP v1 Packets: 49
        CDP v2 Packets: 0
    Invalid CDP Packets: 49
        Unsupported Version: 49
        Checksum Errors: 0
        Malformed Packets: 0

Output Statistics:
```

```
        Total Packets: 47
            CDP v1 Packets: 47
            CDP v2 Packets: 0
    Send Errors: 0
```

Configuration Example for CDP

This example enables the CDP feature and configures the refresh and hold timers:

config t
cdp enable
cdp timer 50
cdp holdtime 100

Additional References

This section includes the following additional information related to CDP:

- Related Documents, page 2-14
- Standards, page 2-14

Related Documents

Related Topic	Document Title
VLAN	*Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.0(4)SV1(3)*
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(3)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for CDP

This section provides the CDP feature release history.

Feature Name	Releases	Feature Information
CDP	4.0(4)SV1(1)	This feature was introduced.

C H A P T E R **3**

Configuring the Domain

This chapter describes how to configure the Cisco Nexus 1000V domain, including creating the domain, assigning VLANs, configuring Layer 3 Control, and so forth.

Information About the Domain

You must create a domain name for Cisco Nexus 1000V and then add control and packet VLANs for communication and management. This process is part of the initial setup of the a Cisco Nexus 1000V when installing the software. If you need to create a domain later, you can do so using the **setup** command or the procedures described in this chapter.

About Layer 3 Control

Layer 3 control, or IP connectivity, is supported between the VSM and VEM for control and packet traffic. With Layer 3 control, a VSM can be Layer 3 accessible and control hosts that reside in a separate Layer 2 network. All hosts controlled by a VSM, however, must still reside in the same Layer 2 network. Since a VSM cannot control a host that is outside of the Layer 2 network it controls, the host on which it resides must be controlled by another VSM.

To implement Layer 3 control, you must make the following configurations:

- Configure the VSM domain transport mode as Layer 3.

  For more information, see the "Changing to Layer 3 Transport" procedure on page 3-5

- Configure a port profile using the "Creating a Port Profile for Layer 3 Control" procedure on page 3-9.

- Create an VMware kernel NIC interface on each host and apply the Layer 3 control port profile to it. For more information, see your VMware documentation.

Figure 3-1 illustrates the following example of Layer 3 control.

- VSM0 controls VEM_0_1.

- VEM_0_1, in turn, hosts VSM1 and VSM2.

- VSM1 and VSM2 control VEMs in other Layer 2 networks.

Figure 3-1   ***Example of Layer 3 Control IP Connectivity***
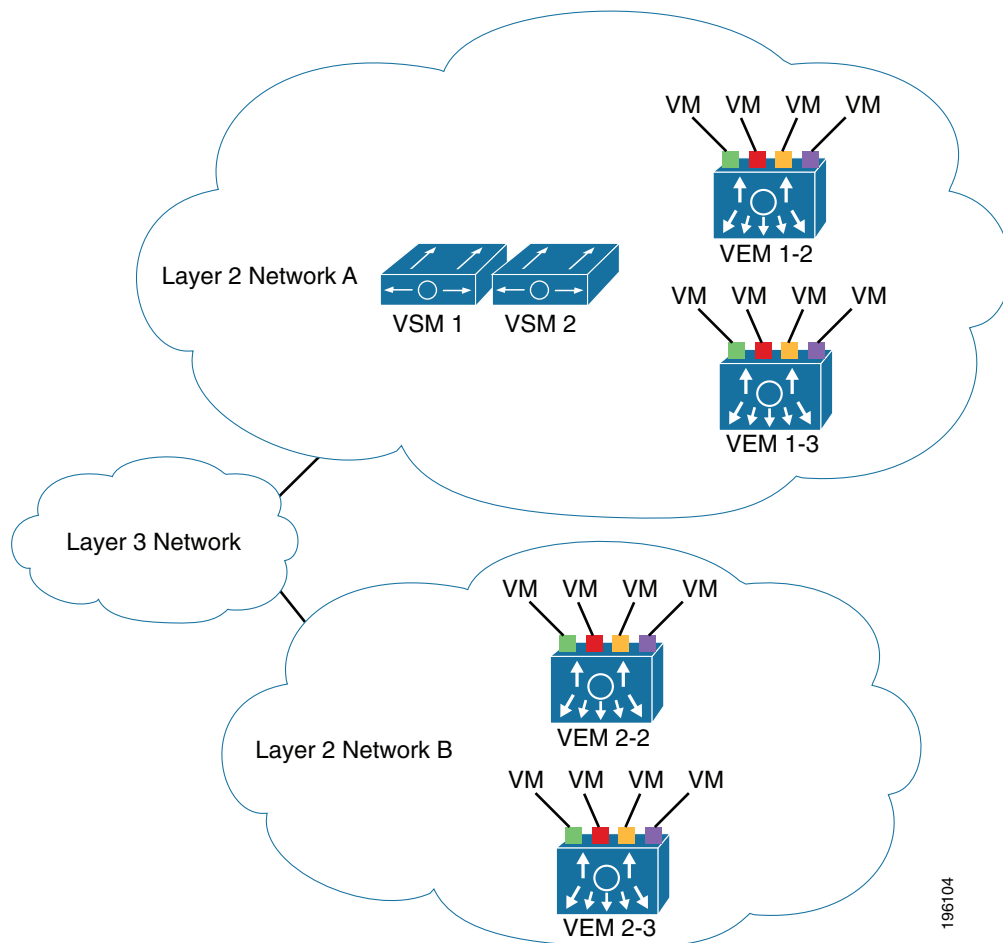
Guidelines and Limitations

The VSM domain has the following configuration guidelines and limitations:

- UDP 4785 is required for Layer 3 communication between the VSM and VEM. If you have a firewall in your network, and are configuring Layer 3 control, then make sure UDP 4785 is open on your upstream switch or firewall device. For more information, see the documentation for your upstream switch or firewall device.

- In a Layer 2 network, you can switch between the Layer 2 and Layer 3 transport modes, but when you do so, the modules may be out of service briefly.

- The capability attribute (Layer 3 control) cannot be inherited from the port profile.

- Different hosts can use different VLANs for Layer 3 control.

- A port profile used for Layer 3 control must be an access port profile. It cannot be a trunk port profile.

- We recommend that if you are using the VMware kernel NIC for Layer 3 Control, you do not use it for any other purpose. For example, do not also use the Layer 3 Control VMware kernel NIC for VMotion or NFS mount.

Default Settings

Table 3-1 lists the default settings in the domain configuration.

Table 3-1   Domain Defaults

Parameter	Default
Control VLAN (svs-domain)	VLAN 1
Packet VLAN (svs-domain)	VLAN 1
VMware port group name (port-profile)	The name of the port profile
SVS mode (svs-domain)	Layer 2
Switchport mode (port-profile)	Access
State (port-profile)	Disabled
State (VLAN)	Active
Shut state (VLAN)	No shutdown

Configuring the Domain

This section includes the following procedures:

Creating a Domain

Use this procedure to create a domain name for the Cisco Nexus 1000V that identifies the VSM and VEMs; and then add control and packet VLANs for communication and management. This process is part of the initial setup of the Cisco Nexus 1000V when installing the software. If you need to create a domain after initial setup, you can do so using this procedure.

BEFORE YOU BEGIN

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

- If two or more VSMs share the same control and/or packet VLAN, the domain helps identify the VEMs managed by each VSM.
- You are logged in to the CLI in EXEC mode.
- You must have a unique domain ID for this Cisco Nexus 1000V instance.
- You must identify the VLANs to be used for control and packet traffic.
- We recommend using one VLAN for control traffic and a different VLAN for packet traffic.

- We recommend using a distinct VLAN for each instances of Cisco Nexus 1000V (different domains)

- The **svs mode** command in the SVS Domain Configuration mode is not used and has no effect on a configuration.

- For information about changing a domain ID after adding a second VSM see the *Cisco Nexus 1000V High Availability and Redundancy Configuration Guide, Release 4.0(4)SV1(3).*

SUMMARY STEPS

1. **config t**
2. **svs-domain**
3. **domain id** *domain-id*
4. **control vlan** *vlan-id*
5. **packet vlan** *vlan-id*
6. **exit**
7. **show svs domain**
8. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**config t** **Example:** n1000v# config t n1000v(config)#	Places you into CLI Global Configuration mode.
Step 2	**svs-domain** **Example:** n1000v(config)# svs-domain n1000v(config-svs-domain)#	Places you into the SVS Domain Configuration mode.
Step 3	**domain id** *number* **Example:** n1000v(config-svs-domain)# domain id 100 n1000v(config-svs-domain)#	Creates the domain ID for this Cisco Nexus 1000V instance.
Step 4	**control vlan** *number* **Example:** n1000v(config-svs-domain)# control vlan 190 n1000v(config-vlan)#	Assigns the control VLAN for this domain.
Step 5	**packet vlan** *number* **Example:** n1000v(config-vlan)# packet vlan 191 n1000v(config-vlan)#	Assigns the packet VLAN for this domain.
Step 6	show svs domain **Example:** n1000v(config-vlan)# show svs domain	Displays the domain configuration.

	Command	Purpose

Step 7	`exit`
	
	Example:
	`n1000v(config-vlan)# exit`
	`n1000v(config)#`
		Returns you to CLI Global Configuration mode.

Step 8	`copy running-config startup-config`
	
	Example:
	`n1000v(config)# copy running-config`
	`startup-config`
		(Optional) Copies the running configuration to the startup configuration.

```
Example:
n1000v# config t
n1000v(config)# svs-domain
n1000v(config-svs-domain)# domain id 100
n1000v(config-svs-domain)# control vlan 190
n1000v(config-svs-domain)# packet vlan 191
n1000v(config-vlan)# exit

n1000v (config)# show svs domain
SVS domain config:
  Domain id:    100
  Control vlan: 190
  Packet vlan:  191
  L2/L3 Aipc mode: L2
  L2/L3 Aipc interface: mgmt0
  Status: Config push to VC successful.

n1000v(config)#
n1000v(config)# copy run start
[#####################################] 100%
n1000v(config)#
```

Changing to Layer 3 Transport

Use this procedure to change the transport mode from Layer 2 to Layer 3 for the VSM domain control and packet traffic.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in EXEC mode.

- This procedure requires you to disable the control and packet VLANs. You cannot change to Layer 3 Control before disabling the control and packet VLANs.

- You have already configured the Layer 3 interface (mgmt 0 or control 0) and assigned an IP address.

  For information about configuring an interface, see the *Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(3)*.

SUMMARY STEPS

1. **show svs domain**

2. **config t**

3. **svs-domain**

4. **no control vlan**

5. **no packet vlan**

6. **show svs domain**

7. **svs mode L2 | svs mode L3 interface { mgmt0 | control0 }**

8. **show svs domain**

9. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**show svs domain**
	
	Example:
	`n1000v(config)# show svs domain`
	`SVS domain config:`
	`  Domain id:    100`
	`  Control vlan: 100`
	`  Packet vlan:  101`
	`  L2/L3 Control mode: L2`
	`  L3 control interface:  NA`
	`  Status: Config push to VC successful.`
		Displays the existing domain configuration, including control and packet VLAN IDs.

Step 2	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 3	**svs-domain**
	
	Example:
	`n1000v(config)# svs-domain`
	`n1000v(config-svs-domain)#`
		Places you in the CLI SVS Domain Configuration mode.

Step 4	**no packet vlan**
	
	Example:
	`n1000v(config-svs-domain)# no packet vlan`
	`n1000v(config-svs-domain)#`
		Removes the packet VLAN configuration.

Step 5	**no control vlan**
	
	Example:
	`n1000v(config-svs-domain)# no control vlan`
	`n1000v(config-svs-domain)#`
		Removes the control VLAN configuration.

Step 6	**show svs domain**
	
	Example:
	`n1000v(config)# show svs domain`
	`SVS domain config:`
	`  Domain id:    100`
	`  Control vlan: 1`
	`  Packet vlan:  1`
	`  L2/L3 Control mode: L2`
	`  L2/L3 Control interface: NA`
	`  Status: Config push to VC successful.`
	`switch(config-svs-domain)#`
		Displays the existing domain configuration, with the default control and packet VLAN IDs.

	Command	Purpose

Step 7	**svs mode L3 interface { mgmt0	control0 }**
		
		Example:
		`n1000v(config-svs-domain)# svs mode l3`
		`interface mgmt0`
		`n000v(config-svs-domain)#`
			Configures Layer 3 transport mode for the VSM domain.
			
			If configuring Layer 3 transport, then you must designate which interface to use; and the interface must already have an IP address configured.
			
			This example shows how to configure Layer 3 transport over the management 0 interface.

Step 8	**show svs domain**
	
	Example:
	`SVS domain config:`
	`  Domain id:    100`
	`  Control vlan: 1`
	`  Packet vlan:  1`
	`  L2/L3 Control mode: L3`
	`  L3 control interface: mgmt0`
	`  Status: Config push to VC successful.`
	`n1000v(config-svs-domain)#`
		(Optional) Displays the new Layer 3 control mode configuration for this VSM domain.

Step 9	**copy running-config startup-config**
	
	Example:
	`n1000v(config-svs-domain)# copy`
	`running-config startup-config`
	`[#######################################]`
	`100%`
	`n1000v(config-svs-domain)#`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Changing to Layer 2 Transport

Use this procedure to change the transport mode to Layer 2 for the VSM domain control and packet traffic. The transport mode is Layer 2 by default, but if it is changed, you can use this procedure to configure it again as Layer 2.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in EXEC mode.

- This procedure requires you to configure a control VLAN and a packet VLAN. You cannot configure these VLANs if the VSM domain capability is Layer 3 Control. You will first change the capability to Layer 3 Control, and then configure the control VLAN and packet VLAN.

SUMMARY STEPS

1. **show svs domain**

2. **config t**

3. **svs-domain**

4. **svs mode L2 | svs mode L3 interface { mgmt0 | control0 }**

5. **show svs domain**

6. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**show svs domain**
	
	Example:
	`SVS domain config:`
	`  Domain id:    100`
	`  Control vlan: 1`
	`  Packet vlan:  1`
	`  L2/L3 Control mode: L3`
	`  L3 control interface: mgmt0`
	`  Status: Config push to VC successful.`
	`n1000v(config-svs-domain)#`
		Displays the existing domain configuration, including control and packet VLAN IDs and the Layer 3 interface configuration.

Step 2	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 3	**svs-domain**
	
	Example:
	`n1000v(config)# svs-domain`
	`n1000v(config-svs-domain)#`
		Places you in the CLI SVS Domain Configuration mode.

Step 4	**svs mode L2**
	
	Example:
	`n1000v(config-svs-domain)# svs mode l2`
	`n000v(config-svs-domain)#`
		Configures Layer 2 transport mode for the VSM domain.

Step 5	**control vlan** *vlanID* **Example:** `n1000v(config-svs-domain)# control vlan 100`	Configures the specified VLAN ID as the control VLAN for the VSM domain.
Step 6	**packet vlan** *vlanID* **Example:** `n1000v(config-svs-domain)# packet vlan 101`	Configures the specified VLAN ID as the packet VLAN for the VSM domain.

Step 7	**show svs domain**
	
	Example:
	`SVS domain config:`
	`  Domain id:    100`
	`  Control vlan: 100`
	`  Packet vlan:  101`
	`  L2/L3 Control mode: L2`
	`  L3 control interface: NA`
	`  Status: Config push to VC successful.`
	`n1000v(config-svs-domain)#`
		(Optional) Displays the new Layer 2 control mode configuration for this VSM domain.

Step 8	**copy running-config startup-config**
	
	Example:
	`n1000v(config-svs-domain)# copy`
	`running-config startup-config`
	`[#######################################]`
	`100%`
	`n1000v(config-svs-domain)#`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Creating a Port Profile for Layer 3 Control

Use this procedure to allow the VSM and VEM to communicate over IP for control and packet traffic.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in EXEC mode.
- The transport mode for the VSM domain has already been configured as Layer 3. For more information, see the "Changing to Layer 3 Transport" procedure on page 3-5.
- All VEMs must belong to the same Layer 2 domain.
- The VEM VM kernel NIC must connect to this Layer 3 control port profile when adding the host to the Cisco Nexus 1000V DVS.
- Only one VM kernel NIC can be assigned to this Layer 3 control port profile per host.
  - If more than one VMware kernel NIC is assigned to a host, the last one assigned takes effect.
  - If more than one VMware kernel NIC is assigned to a host, and you remove the second one assigned, then the VEM does not use the first one assigned. Instead, you must remove both VMware kernel NICs and then add one back.
- You know the VLAN ID for the VLAN you are adding to this Layer 3 control port profile.
  - The VLAN must already be created on the Cisco Nexus 1000V.
  - The VLAN assigned to this Layer 3 control port profile must be a system VLAN.
  - One of the uplink ports must already have this VLAN in its system VLAN range.
- The port profile must be an access port profile. It cannot be a trunk port profile. This procedure includes steps to configure the port profile as an access port profile.
- More than one port profile can be configured as **capability L3 control**.
- Different hosts can use different VLANs for Layer 3 control.
- For additional information about configuring a port profile for ERSPAN using Layer 3 control, see the "Configuring an ERSPAN Port Profile" procedure on page 9-9.

SUMMARY STEPS

1. **config t**
2. **port-profile** *name*
3. **capability l3control**
4. **vmware port-group** [*name*]
5. **switchport mode access**
6. **switchport access vlan** *vlanID*
7. **no shutdown**
8. **system vlan** *vlanID*
9. **state enabled**
10. (Optional) **show port-profile** *name*
11. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	**port-profile** *name*
	
	Example:
	`n1000v(config)# port-profile`
	`l3control-150`
	`n1000v(config-port-prof)#`
		Creates a port profile and places you into Port Profile Configuration mode for the named port profile.
		
		The port profile name can be up to 80 characters and must be unique for each port profile on the Cisco Nexus 1000V.

Step 3	**capability l3control**
	
	Example:
	`n1000v(config-port-prof)# capability`
	`l3control`
	`n1000v(config-port-prof)#`
		Allows the port to be used for IP connectivity.
		
		In vCenter Server, the Layer 3 control port profile must be selected and assigned to the VM kernel NIC physical port.

Step 4	**vmware port-group [**name**]**
	
	Example:
	`n1000v(config-port-prof)# vmware`
	`port-group`
	`n1000v(config-port-prof)#`
		Designates the port-profile as a VMware port group.
		
		The port profile is mapped to a VMware port group of the same name. When a vCenter Server connection is established, the port group created in Cisco Nexus 1000V is then distributed to the virtual switch on the vCenter Server.
		
		name: Port group name. If you do not specify a name, then the port group name will be the same as the port profile name. If you want to map the port profile to a different port group name, use the alternate name.

Step 5	**switchport mode access**]
	
	Example:
	`n1000v(config-port-prof)# switchport`
	`mode access`
	`n1000v(config-port-prof)#`
		Designates that the interfaces are switch access ports (the default).

Step 6	**switchport access vlan** *vlanID*
	
	Example:
	`n1000v(config-port-prof)# switchport`
	`access vlan 150`
	`n1000v(config-port-prof)#`
		Assigns the system VLAN ID to the access port for this Layer 3 control port profile.

Step 7	**no shutdown**
	
	Example:
	`n1000v(config-port-prof)# no shutdown`
	`n1000v(config-port-prof)#`
		Administratively enables all ports in the profile.

Step 8	**system vlan** *vlanID*
	
	Example:
	`n1000v(config-port-prof)# system vlan`
	`150`
	`n1000v(config-port-prof)#`
		Adds the system VLAN to this Layer 3 control port profile.
		
		This ensures that, when the host is added for the first time or rebooted later, the VEM will be able to reach the VSM. One of the uplink ports must have this VLAN in its system VLAN range.

	Command	Purpose

Step 9	**state enabled**
	
	Example:
	`n1000v(config-port-prof)# state enabled`
	`n1000v(config-port-prof)#`
		Enables the Layer 3 control port profile.
		
		The configuration for this port profile is applied to the assigned ports, and the port group is created in the VMware vSwitch on the vCenter Server.

Step 10	**show port-profile name** *name*
	
	Example:
	`n1000v(config-port-prof)# show`
	`port-profile name l3control-150`
	`port-profile l3control-150`
	`  description:`
	`  type: vethernet`
	`  status: enabled`
	`  capability l3control: yes`
	`  pinning control-vlan: 8`
	`  pinning packet-vlan: 8`
	`  system vlans: 150`
	`  port-group: l3control-150`
	`  max ports: 32`
	`  inherit:`
	`  config attributes:`
	`    switchport mode access`
	`    switchport access vlan 150`
	`    no shutdown`
	`  evaluated config attributes:`
	`    switchport mode access`
	`    switchport access vlan 150`
	`    no shutdown`
	`  assigned interfaces:`
	
	`n1000v(config-port-prof)#`
		(Optional) Displays the current configuration for the port profile.

Step 11	**copy running-config startup-config**
	
	Example:
	`n1000v(config-port-prof)# copy`
	`running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Creating a Control VLAN

Use this procedure to add a control VLAN to the domain.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in EXEC mode.
- If Layer 3 Control is configured on your VSM, you can not create a control VLAN. You must first disable Layer 3 Control.
- You have already configured and enabled the required switched virtual interface (SVI) using the document, *Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(3)* The SVI is also called the VLAN interface and provides communication between VLANs.
- You are familiar with how VLANs are numbered. For more information, see the document, *Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.0(4)SV1(3)*.
- Newly-created VLANs remain unused until Layer 2 ports are assigned to them.

SUMMARY STEPS

1. **config t**
2. **vlan** *vlan-id*
3. **name** *vlan-name*
4. **state** *vlan-state*
5. **exit**
6. **show vlan id** *vlan-id*
7. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 2	`vlan 30`
	
	Example:
	`n1000v(config)# vlan 30`
	`n1000v(config-vlan)#`
		Creates VLAN ID 30 for control traffic and places you into CLI VLAN Configuration mode.
		
		Note If you enter a VLAN ID that is assigned to an internally allocated VLAN, the CLI returns an error message.

Step 3	`name cp_control`
	
	Example:
	`n1000v(config-vlan)# name cp_control`
	`n1000v(config-vlan)#`
		Adds the descriptive name, cp_control, to this VLAN.

Step 4	`state active`
	
	Example:
	`n1000v(config-vlan)# state active`
	`n1000v(config-vlan)#`
		Changes the operational state of the VLAN to active.

Step 5	`show vlan id 30`
	
	Example:
	`n1000v(config-vlan)# show vlan id 30`
		Displays the configuration for VLAN ID 30.

Step 6	`copy running-config startup-config`
	
	Example:
	`n1000v(config-vlan)# copy running-config startup-config`
		(Optional) Copies the running configuration to the startup configuration.

```
Example:
n1000v# config t
n1000v(config)# vlan 30
n1000v(config-vlan)# name cp_control
n1000v(config-vlan)# state active
n1000v(config)# show vlan id 30

VLAN Name                         Status    Ports
---- -------------------------------- --------- -------------------------------
```

```
30   cp_control                       active

VLAN Type MTU
---- -----
5    enet 1500

Remote SPAN VLAN
----------------
Disabled

Primary  Secondary  Type           Ports
-------  ---------  -------------- ------------------------------------------

n1000v(config)# copy run start
[#######################################] 100%
n1000v(config)#
```

Creating a Packet VLAN

Use this procedure to add the packet VLAN to the domain.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in EXEC mode.

- You have already configured and enabled the required switched virtual interface (SVI) using the document, *Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(3).* The SVI is also called the VLAN interface and provides communication between VLANs.

- You are familiar with how VLANs are numbered. For more information, see the document, *Cisco Nexus 1000V Layer 2 Switching Configuration Guide, Release 4.0(4)SV1(3)*.

- Newly-created VLANs remain unused until Layer 2 ports are assigned to them.

SUMMARY STEPS

1. **config t**

2. **vlan** *vlan-id*

3. **name** *vlan-name*

4. **state** *vlan-state*

5. **exit**

6. **show vlan id** *vlan-id*

7. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 2	`vlan 31`
	
	Example:
	`n1000v(config)# vlan 31`
	`n1000v(config-vlan)#`
		Creates VLAN ID 31 for packet traffic and places you into CLI VLAN Configuration mode.
		
		Note If you enter a VLAN ID that is assigned to an internally allocated VLAN, the CLI returns an error message.

Step 3	`name cp_packet`
	
	Example:
	`n1000v(config-vlan)# name cp_packet`
	`n1000v(config-vlan)#`
		Adds the descriptive name, cp_packet, to this VLAN.

Step 4	`state active`
	
	Example:
	`n1000v(config-vlan)# state active`
	`n1000v(config-vlan)#`
		Changes the operational state of the VLAN to active.

Step 5	`show vlan id 31`
	
	Example:
	`n1000v(config-vlan)# show vlan id 30`
		Displays the configuration for VLAN ID 31.

Step 6	`exit`
	
	Example:
	`n1000v(config-vlan)# exit`
	`n1000v(config)#`
		Returns you to CLI Global Configuration mode.

Step 7	`copy running-config startup-config`
	
	Example:
	`n1000v(config)# copy running-config startup-config`
		(Optional) Copies the running configuration to the startup configuration.

```
Example:
n1000v# config t
n1000v(config)# vlan 31
n1000v(config-vlan)# name cp_packet
n1000v(config-vlan)# state active
n1000v(config-vlan)# exit
n1000v(config)# show vlan id 31

VLAN Name                             Status      Ports
---- -------------------------------- --------- -------------------------------
31   cp_packet                        active

VLAN Type MTU
---- -----
5    enet 1500

Remote SPAN VLAN
----------------
```

```
                    Disabled

                    Primary  Secondary  Type             Ports
                    -------  ---------  ---------------  ------------------------------------------

                    n1000v(config)# copy run start
                    [#####################################] 100%
                    n1000v(config)#
```

Feature History for the VSM Domain

This section provides the VSM domain feature release history.

Feature Name	Releases	Feature Information
Layer 3 Control	4.0(4)SV1(2)	Added the following information: • About Layer 3 Control, page 3-1 • Guidelines and Limitations, page 3-2 • Changing to Layer 2 Transport, page 3-7 • Changing to Layer 3 Transport, page 3-5 • Creating a Port Profile for Layer 3 Control, page 3-9
VSM Domain	4.0(4)SV1(1)	This feature was introduced.

Managing Server Connections

This chapter describes how to create a connection and connect to a server, how to disconnect from a server, and how to view server connections.

Information About Server Connections

In order to connect to vCenter Server or an ESX server, you must first define the connection in the Cisco Nexus 1000V including the following:

- A connection name
- The protocol used
- The server IP address
- The server DNS name
- The data center name

All communication with vCenter Server is secured by the TLS protocol.

Guidelines and Limitations

Server connections have the following configuration guidelines and limitations:

- A single VSM can only connect to one vCenter server at a time. A single VSM cannot connect to multiple vCenter servers at once.

Connecting to the vCenter Server

Use this procedure to configure a connection and then connect to vCenter server or an ESX server.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- You know the data center name.
- The vCenter Server management station is installed and running.
- You know the vCenter Server IP address or hostname.

- The ESX servers are installed and running.

- The management port is configured.

- The vCenter Server is reachable from the Cisco Nexus 1000V.

- The Cisco Nexus 1000V appliance is installed.

- If you are configuring a connection using a hostname, DNS is already configured.

- You have already registered an extension with the vCenter Server. The extension includes the extension key and public certificate for the VSM. vCenter Server uses these to verify the authenticity of the request it receives from the VSM. For instructions about adding and registering an extension, see the *Cisco Nexus 1000V Software Installation Guide, Release 4.0(4)SV1(3)*.

SUMMARY STEPS

1. **config t**

2. **svs connection** *name*

3. **protocol vmware-vim**

4. **remote {ip address** *address A.B.C.D* **| hostname** *name***}**

5. **vmware dvs datacenter-name** *name*

6. **connect**

DETAILED STEPS

	Command	Description

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 1	`svs connection` *name*
	
	Example:
	`n1000v (config#) svs connection VC`
	`n1000v(config-svs-conn#)`
		Places you into Connection Configuration mode for adding this connection between Cisco Nexus 1000V and either a particular ESX server or the vCenter Server. By using a name, information for multiple connections can be stored in the configuration.

Step 2	`protocol vmware-vim` [http]
	
	Example:
	`n1000v(config-svs-conn#) protocol vmware-vim`
	`n1000v(config-svs-conn#)`
		Specifies that this connection uses the VIM protocol. This command is stored locally.
		
		- **http**: Specifies that the VIM protocol runs over HTTP. The default is to use HTTP over SSL (HTTPS).

Step 3	Do one of the following: - If you are configuring an IP address, go to Step 4. - If you are configuring a hostname, go to Step 5.	

	Command	Description

Step 4	`remote ip address` *ipaddress*
	
	Example:
	`n1000v(config-svs-conn#) remote ip address`
	`192.168.0.1`
	`n1000v(config-svs-conn#)`
	
	Go to Step 6.
		Specifies the IP address of the ESX server or vCenter Server for this connection. This command is stored locally.

Step 5	`remote hostname` *hostname*
	
	Example:
	`n1000v(config-svs-conn#) remote hostname vcMain`
	`n1000v(config-svs-conn#)`
		Specifies the DNS name of the ESX server or vCenter Server for this connection. This command is stored locally.
		
		Note DNS is already configured.

Step 6	`vmware dvs datacenter-name` *name*
	
	Example:
	`n1000v(config-svs-conn#) vmware dvs datacenter-name`
	`Hamilton-DC`
	`n1000v(config-svs-conn#)`
		Identifies the data center name in the vCenter Server where Cisco Nexus 1000V is to be created as a distributed virtual switch (DVS). You can use this command before or after connecting. The data center name is stored locally.

Step 7	`connect`
	
	Example:
	`n1000v(config-svs-conn#) connect`
		Initiates the connection. If the username and password have not been configured for this connection, the user is prompted for a username and password.
		
		The default is no connect. There can be only one active connection at a time. If a previously-defined connection is up, an error message displays and the command is rejected until the user closes the previous connection by entering no connect.

```
Example:
n1000v# config t
n1000v (config)# svs connection VC
n1000v(config-svs-conn#) protocol vmware-vim
n1000v(config-svs-conn#) remote ip address 192.168.0.1
n1000v(config-svs-conn#) vmware dvs datacenter-name Hamilton-DC
n1000v(config-svs-conn#) connect


n1000v# show svs connections
connection VC:
    ip address: 192.168.0.1
    protocol: vmware-vim https
    certificate: default
    datacenter name: Hamilton-DC
    DVS uuid: ac 36 07 50 42 88 e9 ab-03 fe 4f dd d1 30 cc 5c
    config status: Enabled
    operational status: Connected
n1000v#
```

Disconnecting From the vCenter Server

Use this procedure to disconnect from the vCenter Server, for example, after correcting a vCenter Server configuration

BEFORE YOU BEGIN

- You are logged in to the Cisco Nexus 1000V in EXEC mode.
- You have configured an Cisco Nexus 1000V connection using the "Connecting to the vCenter Server" procedure on page 4-1.
- The Cisco Nexus 1000V is connected to vCenter Server/ESX.

DETAILED STEPS

	Command	Description

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 1	`svs connection` *name*
	
	Example:
	`n1000v (config#) svs connection vcWest`
	`n1000v(config-svs-conn)#`
		Places you into a CLI Global Configuration submode for the connection to vCenter Server.

Step 2	`no connect`
	
	Example:
	`n1000v(config-svs-conn)# no connect`
	`n1000v(config-svs-conn)#`
		Closes the connection.

Removing the DVS from the vCenter Server

Use this procedure to remove the DVS from the vCenter Server.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- You have configured a connection using the "Connecting to the vCenter Server" procedure on page 4-1.
- The Cisco Nexus 1000V is connected to vCenter Server/ESX.
- The Server Administrator has already removed from the VI client all of the hosts connected to Cisco Nexus 1000V. For more information, see the VMware documentation.

DETAILED STEPS

	Command	Description

Step 1	`config t`
	
	`Example:`
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 1	`svs connection` *name*
	
	`Example:`
	`n1000v(config#) svs connection vcWest`
	`n1000v(config-svs-conn)#`
		Places you into a CLI Global Configuration submode for the connection to vCenter Server.

Step 2	`no vmware dvs`
	
	`Example:`
	`n1000v(config-svs-conn)# no vmware dvs`
	`n1000v(config-svs-conn)#`
		Removes the DVS associated with the specified connection from the vCenter Server.

Configuring Host Mapping

This section includes the following topics:

Information about Host Mapping

When a VSM detects a new VEM, it automatically assigns a free module number to the VEM and then maintains the mapping between the module number and UUID of a host server. This mapping is used to assign the same module number to a given host server.

Removing Host Mapping from a Module

Use this procedure to remove the mapping of a module to a host server.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- You have already removed the host from the Cisco Nexus 1000V DVS on vCenter.

SUMMARY STEPS

1. **config t**
2. **no vem** *module-number*

3. **show module vem mapping**

4. **copy running-config startup-config**

DETAILED STEPS

	Command	Description

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 1	`no vem` *module-number*
	
	Example:
	`n1000v(config)# no vem 4`
	`n1000v(config)# no vem 3`
	`cannot modify slot 3: host module is inserted`
	`n1000v((config)#`
		Removes the specified module from software.
		
		Note If the module is still present in the slot, the command is rejected, as shown in this example.

Step 2	`show module vem mapping`
	
	Example:
	`n1000v(config)# show module vem mapping`
	`Mod     Status        UUID                                 License Status`
	`---     ----------    -----------------------------------  --------------`
	`  3     powered-up    93312881-309e-11db-afa1-0015170f51a8   licensed`
	`n1000v(config)#`
		(Optional) Displays the mapping of modules to host servers.

Step 3	`copy running-config startup-config`
	
	Example:
	`n1000v(config-vem-slot)# copy running-config`
	`startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Mapping to a New Host

Use this procedure to map a module number to a different host server UUID.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.

- You have already removed the host from the Cisco Nexus 1000V DVS on vCenter using the "Removing Host Mapping from a Module" procedure on page 4-5.

Note   If you don't first remove the existing host server mapping, the new host server is assigned a different module number.

SUMMARY STEPS

1. **config t**

2. **vem module** *number*

3. **host vmware id** *server-bios-uuid*

4. **show module vem mapping**

5. **copy running-config startup-config**

DETAILED STEPS

	Command	Description

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 1	`vem` *module number*
	
	Example:
	`n1000v(config)# vem 3`
	`n1000v((config-vem-slot)#`
		Places you into CLI VEM Slot Configuration mode.

Step 2	`host vmware id` *server-bios-uuid*
	
	Example:
	`n1000v(config-vem-slot)# host vmware id`
	`6dd6c3e3-7379-11db-abcd-000bab086eb6`
	`n1000v(config-vem-slot)#`
		Assigns a different host server UUID to the specified module.

Step 3	`show module vem mapping`
	
	Example:
	`n1000v(config-vem-slot)# show module vem mapping`
	`Mod    Status        UUID                                 License Status`
	`---    ----------    ----------------------------------   --------------`
	`  3    powered-up    93312881-309e-11db-afa1-0015170f51a8   licensed`
	`  4       absent    6dd6c3e3-7379-11db-abcd-000bab086eb6   licensed`
	
	`n1000v(config-vem-slot)#`
		(Optional) Displays the mapping of modules to host servers.

Step 4	`copy running-config startup-config`
	
	Example:
	`n1000v(config-vem-slot)# copy running-config`
	`startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Viewing Host Mapping

Use this procedure in EXEC mode to view the mapping of modules to host servers.

Step 1 **show module vem mapping**

Example:
```
n1000v(config)# show module vem mapping
Mod    Status        UUID                                 License Status

---    ----------    ----------------------------------   --------------

  3    powered-up    93312881-309e-11db-afa1-0015170f51a8   licensed
```

```
n1000v(config)#
```

Verifying Connections

Use this procedure to view and verify connections.

BEFORE YOU BEGIN

- You are logged in to the CLI in any command mode.
- You have configured the connection using the "Connecting to the vCenter Server" procedure on page 4-1.
- The Cisco Nexus 1000V is connected to vCenter Server/ESX.

DETAILED STEPS

	Command	Description

Step 1	`show svs connections` [*name*]
	
	Example:
	`n1000v# show svs connections vc`
	`Connection vc:`
	`IP address: 172.28.15.206`
	`Protocol: vmware-vim https`
	`vmware dvs datacenter-name: HamiltonDC`
	`ConfigStatus: Enabled`
	`OperStatus: Connected`
	`n1000v#`
		Displays the current connections to the Cisco Nexus 1000V.
		
		Note Network connectivity issues may shut down your connection to the vCenter Server. When network connectivity is restored, the Cisco Nexus 1000V will not automatically restore the connection. In this case, you must restore the connection manually using the following command sequence,
		
		`no connect`
		`connect`

Verifying the Domain

Use this procedure to view and verify the configured domain.

BEFORE YOU BEGIN

- You are logged in to the CLI in any command mode.
- You have configured a domain using the "Creating a Domain" procedure on page 3-3.

DETAILED STEPS

	Command	Description

Step 1	`show svs domain`
	
	Example:
	n1000v# **show svs domain**
	SVS domain config:
	  Domain id: 98
	  Control vlan: 70
	  Packet vlan: 71
	  Sync state: -
	
	n1000v#
		Display the domain configured on the Cisco Nexus 1000V.

Verifying the Configuration

Use this procedure to display and verify the running configuration.

BEFORE YOU BEGIN

- You are logged in to the CLI in any command mode.

- You have configured Cisco Nexus 1000V connections using the "Connecting to the vCenter Server" procedure on page 4-1.

- The Cisco Nexus 1000V is connected to vCenter Server/ESX.

DETAILED STEPS

	Command	Description
Step 1	`show running-config`	Display the current configuration. If the Cisco Nexus 1000V is not connected to a vCenter Server or ESX server, the output is limited to connection-related information.

```
Example:
n1000v(config-acl)# show running-config
version 4.0(4)SV1(1)
feature port-security
username adminbackup password 5 $1$Oip/C5Ci$oOdx7oJSlBCFpNRmQK4na.  role network-operator
username admin password 5 $1$N1mX5tLD$daXpuxlAPcIHoz53PBhy6/  role network-admin
telnet server enable
ssh key rsa 1024 force
kernel core target 0.0.0.0
kernel core limit 1
system default switchport
ip access-list my66
  10 permit ip 1.1.1.1/32 1.1.1.2/32
```

```
snmp-server user admin network-admin auth md5 0x90f3798f3e894496a11ec42ce2efec9c priv
0x90f3798f3e894496a11ec42ce2efec9c localizedkey
snmp-server enable traps entity fru
snmp-server enable traps license
vrf context management
  ip route 0.0.0.0/0 172.28.15.1
switchname srini-cp
vlan 40-43,45-48
vdc srini-cp id 1
  limit-resource vlan minimum 16 maximum 4094
  limit-resource monitor-session minimum 0 maximum 32
  limit-resource vrf minimum 16 maximum 8192
  limit-resource port-channel minimum 0 maximum 192
  limit-resource u4route-mem minimum 32 maximum 256
  limit-resource u6route-mem minimum 16 maximum 256

interface Ethernet6/2
  inherit port-profile uplinkportprofile1

interface Ethernet6/3
  inherit port-profile uplinkportprofile2

interface Ethernet6/4
  inherit port-profile uplinportprofile3

interface Ethernet7/2
  inherit port-profile uplinkportprofile1

interface mgmt0
  ip address 172.28.15.163/24

interface Vethernet1


  inherit port-profile vm100

interface Vethernet2


  inherit port-profile vm100

interface Vethernet3


  inherit port-profile vm100

interface Vethernet4


  inherit port-profile vm100

interface Vethernet5

interface Vethernet6
boot kickstart bootflash:/svs-kickstart-mzg.4.0.1a.S1.0.82.bin sup-1
boot system bootflash:/svs-mzg.4.0.1a.S1.0.82.bin sup-1
boot system bootflash:/isan.bin sup-1
boot kickstart bootflash:/svs-kickstart-mzg.4.0.1a.S1.0.82.bin sup-2
boot system bootflash:/svs-mzg.4.0.1a.S1.0.82.bin sup-2
boot system bootflash:/isan.bin sup-2
ip route 0.0.0.0/0 172.28.15.1
port-profile uplinkportprofile1
  capability uplink
  vmware port-group
```

```
        switchport mode trunk
        switchport trunk allowed vlan 1,40-43
        no shutdown
        system vlan 1,40-43
        state enabled
      port-profile vm100
        vmware port-group
        switchport mode access
        switchport access vlan 43
        ip port access-group my100 out
        ip port access-group my66 in
        no shutdown
        state enabled
      port-profile uplinkportprofile2
        capability uplink
        vmware port-group
        switchport mode trunk
        switchport trunk allowed vlan 45-46
        no shutdown
        state enabled
      port-profile uplinportprofile3
        capability uplink
        vmware port-group
        switchport trunk allowed vlan 47-48
        state enabled
      port-profile uplinkportprofile3
        no shutdown
      svs-domain
        domain id 163
        control vlan 41
        packet vlan 42
      svs connection VCR5
        protocol vmware-vim
        remote ip address 172.28.30.83
        vmware dvs datacenter-name cisco-DC
        connect
      n1000v(config-acl)#
```

Verifying Module Information

Use this procedure to display and verify module information, including a view of the DVS from Cisco Nexus 1000V.

BEFORE YOU BEGIN

- You are logged in to the CLI in any command mode.

- You have configured the Cisco Nexus 1000V connection using the "Connecting to the vCenter Server" procedure on page 4-1.

- The Cisco Nexus 1000V is connected to vCenter Server/ESX.

- The Server Administrator has already added the host running Cisco Nexus 1000V to the DVS in vCenter Server.

SUMMARY STEPS

1. **show module**

2. **show server-info**

   3. **show interface brief**

   4. **show interface virtual**

DETAILED STEPS

	Command	Description

Step 1	`show module`
	
	Example:
	`n1000v# show module`
		Displays module information.

Step 2	`show server_info`
	
	Example:
	`n1000v# show server_info`
		Displays server information.

Step 3	`show interface brief`
	
	Example:
	`n1000v# show interface brief`
		Displays interface information, including the uplinks to vCenter Server.

Step 4	`show interface virtual`
	
	Example:
	`n1000v# show interface virtual`
		Displays virtual interface information.

```
Example
n1000v# show  module
Mod   Ports  Module-Type                        Model             Status
---   -----  ------------------------------  ----------------  -----------
1     1      Virtual Supervisor Module         Nexus1000V        active *
2     48     Virtual Ethernet Module                             ok
3     48     Virtual Ethernet Module                             ok


Mod  Sw             Hw      World-Wide-Name(s) (WWN)
---  -------------- ------  --------------------------------------------------
1    4.0(0)S1(0.82) 0.0     --
2    NA             0.0     --
3    NA             0.0     --


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


Mod  Server-IP       Server-UUID                         Server-Name
---  --------------  ----------------------------------  --------------------
1    172.18.217.180                                      esx-1
2    172.18.117.44   487701ee-6e87-c9e8-fb62-001a64d20a20  esx-2
3    172.18.217.3    4876efdd-b563-9873-8b39-001a64644a24  esx-3


* this terminal session



Example
n1000v# show  server_info

Mod      Status          UUID
---      ----------      ----
```

```
     2      powered-up      34303734-3239-5347-4838-323130344654
     3      absent               371e5916-8505-3833-a02b-74a4122fc476
     4      powered-up      4880a7a7-7b51-dd96-5561-001e4f3a22f9
     5      absent               48840e85-e6f9-e298-85fc-001e4f3a2326
     6      powered-up      eb084ba6-3b35-3031-a6fe-255506d10cd0
n1000v#
```

Example
n1000v# **show interface brief**

```
--------------------------------------------------------------------------------
Port    VRF          Status IP Address                          Speed    MTU
--------------------------------------------------------------------------------
mgmt0   --           up     172.28.15.211                       1000     1500

--------------------------------------------------------------------------------
Ethernet    VLAN   Type Mode    Status   Reason                 Speed    Port
Interface                                                                Ch #
--------------------------------------------------------------------------------
Eth2/2      1      eth  trunk   up       none                   a-1000(D) --

--------------------------------------------------------------------------------
Interface   VLAN   Type Mode    Status   Reason                 MTU
--------------------------------------------------------------------------------
```

Example
n1000v# **show interface virtual**

```
--------------------------------------------------------------------------------
Port        Adapter       Owner               Mod Host
--------------------------------------------------------------------------------
Veth49                    R-VM-1              2   mcs-srvr35
```

Feature History for Server Connections

This section provides the server connections feature release history.

Feature Name	Releases	Feature Information
Server Connections	4.0(4)SV1(1)	This feature was introduced.

C H A P T E R **5**

Managing the Configuration

Information About Configuration Management

The Cisco Nexus 1000V provides you with the capability to change the switch name, configure messages of the day, and display, save, and erase configuration files.

Changing the Switch Name

Use this procedure to change the switch name or prompt from the default (switch#) to another character string.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in configuration mode.
- If the VSM is connected to vCenter Server then this procedure also changes the DVS the VSM is managing.In case of error in renaming the DVS, a syslog is generated and the DVS on vCenter Server will continue using the old DVS name.

DETAILED STEPS

	Command	Purpose

Step 1	`switchname`
	
	Example:
	`n1000v(config)# switchname metro`
	`metro(config)# exit`
	`metro#`
		Changes the switch prompt.

Configuring a Message of the Day

Use this procedure to configure a message of the day (MOTD) to display before the login prompt on the terminal when a user logs in.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in configuration mode.
- The banner message can be up to 40 lines with up to 80 characters per line.
- Use the following guidelines when choosing your delimiting character:
  - Do not use the *delimiting-character* in the *message* string.
  - Do not use **"** and **%** as delimiters.
- The following tokens can be used in the message of the day:
  - `$(hostname)` displays the host name for the switch.
  - `$(line)` displays the vty or tty line or name.

DETAILED STEPS

	Command	Purpose

Step 1	**banner motd [**_delimiting-character message delimiting-character_**]**
	
	Example:
	`n1000v(config)# banner motd #April 16, 2008 Welcome to the svs#`
	`n1000v(config)#`
		Configures a banner message of the day.
		
		• up to 40 lines
		• up to 80 characters per line
		• enclosed in delimiting character, such as #
		• can span multiple lines
		• can use tokens

Step 2	**show banner motd**
	
	Example:
	`n1000v(config)# show banner motd`
	`April 16, 2008 Welcome to the Switch`
		Displays the configured banner message.

Verifying the Configuration

Use this section to view the switch configuration. This section includes the following topics:

Verifying the Software and Hardware Versions

Use this command to view the versions of software and hardware on your system, for example, to verify the version before and after an upgrade.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI in any command mode.

DETAILED STEPS

	Command	Description

Step 1	`show version`
	
	Example:
	`n1000v# show version`
		Displays the versions of system software and hardware that are currently running on the switch,

```
Example:
n1000v# show version
Cisco Nexus Operating System (NX-OS) Software
TAC support: http://www.cisco.com/tac
Copyright (c) 2002-2009, 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

Software
  loader:    version 1.2(2)
  kickstart: version 4.0(4)SV1(1)
  system:    version 4.0(4)SV1(1)
  kickstart image file is:
  kickstart compile time:  4/2/2009 23:00:00
  system image file is:    bootflash:/svs.bin
  system compile time:     4/2/2009 23:00:00 [04/23/2009 09:55:29]


Hardware
  Cisco Nexus 1000V Chassis ("Virtual Supervisor Module")
  Intel(R) Xeon(R) CPU        with 1034780 kB of memory.
  Processor Board ID T5056893321

  Device name: n1000v
  bootflash:    3897832 kB

Kernel uptime is 0 day(s), 0 hour(s), 2 minute(s), 55 second(s)


plugin
  Core Plugin, Ethernet Plugin
```

Verifying the Running Configuration

Use this section to view the configuration currently running on the system.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI in any command mode.

DETAILED STEPS

	Command	Description

Step 1	`show running-config`
	
	Example:
	`n1000v# show running-config`
		Displays the versions of system software and hardware that are currently running on the switch,

```
Example:
n1000v# show running-config
version 4.0(4)SV1(1)
username admin password 5 $1$ouYE/pRM$/j4/2lg3RMd4PhE.1Z1S.0  role network-admin
telnet server enable
ip domain-lookup
ip host n1000v 172.23.232.141
kernel core target 0.0.0.0
kernel core limit 1
system default switchport
vem 3
  host vmware id 89130a67-e66b-3e57-ad25-547750bcfc7e
snmp-server user admin network-admin auth md5 0xb64ad6879970f0e57600c443287a79f0 priv
0xb64ad6879970f0e57600c443287a79f0 localizedkey
snmp-server enable traps license
vrf context management
  ip route 0.0.0.0/0 172.23.232.1
switchname n1000v
vlan 1,260-269
vdc n1000v id 1
  limit-resource vlan minimum 16 maximum 513
  limit-resource monitor-session minimum 0 maximum 64
  limit-resource vrf minimum 16 maximum 8192
  limit-resource port-channel minimum 0 maximum 256
  limit-resource u4route-mem minimum 32 maximum 80
  limit-resource u6route-mem minimum 16 maximum 48
port-profile Unused_Or_Quarantine_Uplink
  description "Port-group created for Nexus1000V internal usage. Do not use."
  capability uplink
  vmware port-group
  shutdown
  state enabled
port-profile Unused_Or_Quarantine_Veth
  description "Port-group created for Nexus1000V internal usage. Do not use."
  vmware port-group
  shutdown
  state enabled
port-profile system-uplink
  capability uplink
  vmware port-group
  switchport mode trunk
```

```
  switchport trunk allowed vlan 260-261
  no shutdown
  system vlan 260-261
  state enabled
port-profile vm-uplink
  capability uplink
  vmware port-group
  switchport mode access
  switchport access vlan 262
  no shutdown
  state enabled
port-profile data262
  vmware port-group
  switchport access vlan 262
  no shutdown
  state enabled

interface Ethernet3/2
  inherit port-profile system-uplink

interface Ethernet3/3
  inherit port-profile vm-uplink

interface mgmt0
  ip address 172.23.232.141/24

interface control0
line vty
  session-limit 32
boot kickstart bootflash:/kick.bin sup-1
boot system bootflash:/svs.bin sup-1
boot kickstart bootflash:/kick.bin sup-2
boot system bootflash:/svs.bin sup-2
svs-domain
  domain id 141
  control vlan 260
  packet vlan 261
  svs mode L2
svs connection vc
  protocol vmware-vim
  remote hostname 172.23.231.201
  vmware dvs uuid "2c 6f 3d 50 62 f3 7f 4d-dc 00 70 e2 52 77 ca 15" datacenter-name
HamiltonDC
  connect

n1000v#
```

Comparing the Startup and Running Configurations

Use this procedure to view the difference between the startup and running configurations.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI in any command mode.

DETAILED STEPS

	Command	Description
Step 1	show running-config diff **Example:** n1000v# show running-config diff	Displays the difference between the startup configuration and the running configuration currently on the switch.

Example 5-1   Command output, `show running-config diff`

```
n1000v# show running-config diff
*** Startup-config
--- Running-config
***************
*** 1,7 ****
  version 4.0(1)
- system mem-thresholds minor 0 severe 0 critical 0
  vrf context management
    ip route 0.0.0.0/0 10.78.1.1
  switchname DCOS-112-S10
  vlan 80,110-111,150,160,170
  vdc DCOS-112-S10 id 1
--- 1,6 ----
***************
*** 116,131 ****
    ip address 10.78.1.112/24
  interface Vethernet49
    inherit port-profile vlan160
- interface Vethernet65
-   inherit port-profile vlan170
  interface Vethernet50
    inherit port-profile vlan160
  interface Vethernet66
    inherit port-profile vlan170
  ip route 0.0.0.0/0 10.78.1.1
  vlan 80-80, 110-110, 111-111, 150-150, 160-160, 170-170

--- 115,130 ----
    ip address 10.78.1.112/24

  interface Vethernet49
    inherit port-profile vlan160

  interface Vethernet50
    inherit port-profile vlan160

+ interface Vethernet65
+   inherit port-profile vlan170
+
  interface Vethernet66
    inherit port-profile vlan170
  ip route 0.0.0.0/0 10.78.1.1
  vlan 80-80, 110-110, 111-111, 150-150, 160-160, 170-170

n1000v#
```

Verifying the Interface Configuration

This section includes the following procedures:

For more information about displaying interfaces, see the document, *Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(3)*

Verifying a Brief Version of an Interface Configuration

Use this procedure to view a brief version of an interface configuration.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in any command mode.

DETAILED STEPS

	Command	Description
Step 1	**show interface** {*type*} {*name*} **brief**	Displays a brief version of information about the specified interface configuration,

```
Example:
n1000v# show interface mgmt 0 brief

--------------------------------------------------------------------------------
Port   VRF        Status IP Address                              Speed    MTU
--------------------------------------------------------------------------------
mgmt0  --         up     10.78.1.63                              1000     1500
n1000v#
```

Verifying a Detailed Version of an Interface Configuration

Use this procedure to view a detailed version of an interface configuration.

BEFORE YOU BEGIN

Before using the commands in this section, you must know or do the following:

- You are logged in to the CLI in any command mode.

DETAILED STEPS

	Command	Description
Step 1	`show interface {type} {name}`	Displays details about the specified interface configuration,

```
Example:
n1000v# show interface mgmt 0
mgmt0 is up
  Hardware: Ethernet, address: 0050.5689.3321 (bia 0050.5689.3321)
  Internet Address is 172.23.232.141/24
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA
  full-duplex, 1000 Mb/s
  Auto-Negotiation is turned on
    4961 packets input, 511995 bytes
    0 multicast frames, 0 compressed
    0 input errors, 0 frame, 0 overrun, 0 fifo
    245 packets output, 35853 bytes
    0 underrun, 0 output errors, 0 collisions
    0 fifo, 0 carrier errors

n1000v#
```

Verifying a Brief Version of all Interfaces

Use this procedure to view a brief version of all interfaces configured on your system.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in any command mode.

DETAILED STEPS

	Command	Description
Step 1	`show interface brief`	Displays a brief version of all interface configurations on your system,

Example:
```
n1000v# show interface brief

--------------------------------------------------------------------------------
Port     VRF          Status IP Address                       Speed    MTU
--------------------------------------------------------------------------------
mgmt0    --           up     172.23.232.141                   1000     1500


--------------------------------------------------------------------------------
Ethernet     VLAN  Type Mode    Status Reason                  Speed     Port
Interface                                                                 Ch #
--------------------------------------------------------------------------------
Eth3/2       1     eth  trunk   up     none                     1000(D) --
Eth3/3       262   eth  access  up     none                     1000(D) --
--------------------------------------------------------------------------------
Interface    VLAN  Type Mode    Status Reason                  MTU
--------------------------------------------------------------------------------
Veth81       630   virt access  up     none                    1500
Veth82       630   virt access  up     none                    1500
Veth224      631   virt access  up     none                    1500
Veth225      1     virt access  nonPcpt nonParticipating       1500
n1000v#
```

Verifying the Running Configuration for all Interfaces

Use this procedure to view the running configuration for all interfaces on your system.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in any command mode.
- The output for the command, **show running-config interface** differs from that of the command, **show interface**.

DETAILED STEPS

	Command	Description
Step 1	show running-config interface	Displays the running configuration for all interfaces on your system,

Example:
```
n1000v# show running-config interface
version 4.0(1)

interface Ethernet3/2
  switchport
  inherit port-profile sftrunk

interface Ethernet3/6
  switchport
  inherit port-profile vmuplink

interface Ethernet6/2
  switchport
  inherit port-profile alluplink
```

```
interface mgmt0
  ip address 10.78.1.63/24

interface Vethernet81
  inherit port-profile vm630

interface Vethernet82
  inherit port-profile vm630

interface Vethernet224
  inherit port-profile vm631

interface Vethernet225

n1000v#
```

Saving a Configuration

Use this procedure to save the running configuration to the startup configuration so that your changes are retained in the configuration file the next time you start the system.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI in any command mode.

DETAILED STEPS

	Command	Description
Step 1	**copy running-config startup-config**	Saves the new configuration into nonvolatile storage, after which the running and the startup copies of the configuration are identical.

```
Example:
n1000v(config)# copy run start
[#####################################] 100%
n1000v(config)#
```

Erasing a Configuration

Use this procedure to erase a startup configuration.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

⚠

Caution   The **write erase** command erases the entire startup configuration with the exception of loader functions, the license configuration, and the certificate extension configuration.

- You are logged in to the CLI.
- The following parameters are used with this command:
  - boot: Erases the boot variables and the mgmt0 IP configuration.
  - debug: Erases the debug configuration.

DETAILED STEPS

	Command	Description
Step 1	`write erase` [boot | debug]	The existing startup configuration is completely erased and all settings revert to their factory defaults.
		The running configuration is not affected.

Feature History for Configuration Management

This section provides the configuration management feature release history.

Feature Name	Releases	Feature Information
Configuration Management	4.0(4)SV1(1)	This feature was introduced.

C H A P T E R **6**

Working with Files

Information About Files

The Cisco Nexus 1000V file system provides a single interface to all the file systems the switch uses, including:

- Flash memory file systems
- Network file systems (TFTP and FTP)
- Any other endpoint for reading or writing data (such as the running configuration)

Navigating the File System

This section describes how to navigate the file system.

Specifying File Systems

The syntax for specifying a file system is `<file system name>:[//server/]`. Table 6-1 describes file system syntax.

Table 6-1   File System Syntax Components

File System Name	Server	Description
bootflash	sup-active sup-local sup-1 module-1	Internal memory located on the active supervisor used for storing system images, configuration files, and other miscellaneous files. Cisco Nexus 1000V CLI defaults to the bootflash: file system.
	sup-standby sup-remote sup-2 module-2	Internal memory located on the standby supervisor used for storing system images, configuration files, and other miscellaneous files.
volatile	—	Volatile random-access memory (VRAM) located on a supervisor module used for temporary or pending changes.

Identifying the Directory You are Working From

Use this procedure to display the directory name of your current CLI location.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI.

DETAILED STEPS

Step	Command	Purpose

Step 1	`pwd`
	
	Example:
	`n1000v# pwd`
	`bootflash:`
		Displays the present working directory.

Changing Your Directory

Use this procedure to change your location in the CLI, from one directory or file system to another.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI in any command mode.
- Cisco Nexus 1000V CLI defaults to the bootflash: file system.

🔍

Tip   Any file saved in the volatile: file system is erased when the switch reboots.

DETAILED STEPS

Step	Command	Purpose

Step 1	`pwd`
	
	Example:
	`n1000v# pwd`
	`volatile:`
	`n1000v#`
		Displays the directory name of your current CLI location.

Step	Command	Purpose
Step 2	**cd** *directory name*	Changes your CLI location to the specified directory.
	Example: n1000v# cd bootflash:	Changes your CLI location to the root directory on the bootflash: file system.
	Example: n1000v# cd bootflash:mydir	Changes your CLI location to the mydir directory that resides in the bootflash: file system.
	Example: n1000v# cd mystorage	Changes your CLI location to the mystorage directory that resides within the current directory. If the current directory were bootflash: mydir, this command changes the current directory to bootflash: mydir/mystorage.

Listing the Files in a File System

Use this procedure to display the contents of a directory or file.

DETAILED STEPS

Step	Command	Purpose
Step 1	**dir** [*directory*	*filename*]	Displays the contents of a directory or file.

```
Example:
DCOS-112-R5# dir lost+found/
      49241    Jul 01 09:30:00 2008  diagclient_log.2613
      12861    Jul 01 09:29:34 2008  diagmgr_log.2580
         31    Jul 01 09:28:47 2008  dmesg
       1811    Jul 01 09:28:58 2008  example_test.2633
         89    Jul 01 09:28:58 2008  libdiag.2633
      42136    Jul 01 16:34:34 2008  messages
         65    Jul 01 09:29:00 2008  otm.log
        741    Jul 01 09:29:07 2008  sal.log
         87    Jul 01 09:28:50 2008  startupdebug

Usage for log://sup-local
   51408896 bytes used
  158306304 bytes free
  209715200 bytes total
DCOS-112-R5#
```

Identifying Available File Systems for Copying Files

Use this procedure to identify the file systems you can copy to or from.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in EXEC mode.

DETAILED STEPS

Step	Command	Purpose
Step 1	`copy ?`	Displays the source file systems available to the copy command.

Step 2	`copy filename ?`
	
	Example:
	`n1000v# copy ?`
	`bootflash: Select source filesystem`
	`core: Select source filesystem`
	`debug: Select source filesystem`
	`ftp: Select source filesystem`
	`licenses Backup license files`
	`log: Select source filesystem`
	`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`
		Displays the destination file systems available to the copy command for a specific file.

Using Tab Completion

Use this procedure to have the CLI complete a partial file name in a command.

	Command	Purpose

Step 1	`show file` *filesystem name: partial filename* `<Tab>`
	
	Example:
	`n1000v# show file bootflash:nexus-1000v-`
	`bootflash:nexus-1000v-dplug-mzg.4.0.4.SV1.`
	`0.42.bin`
	`bootflash:nexus-1000v-mzg.4.0.4.SV1.0.42.b`
	`in`
	`bootflash:nexus-1000v-kickstart-mzg.4.0.4.`
	`SV1.0.42.bin`
		When you type a partial filename and then press Tab, the CLI completes the file name if the characters you typed are unique to a single file.
		
		If not, the CLI lists a selection of file names that match the characters you typed.
		
		You can then retype enough characters to make the file name unique; and CLI completes the file name for you.

Step 2	`show file bootflash:c` `<Tab>`
	
	Example:
	`n1000v# show file bootflash:c<Tab>`
	`-----BEGIN RSA PRIVATE KEY-----`
	`MIICXgIBAAKBgQDSq93BrlHcg3bX1jXDMY5c9+yZSS`
	`T3VhuQBqogvCPDGeLecA+j`
	`...`
	`...`
	`n1000v#`
		The CLI completes the file name for you.

Copying and Backing Up Files

Use this procedure to copy a file, such as a configuration file, to save it or reuse it at another location. If your internal file systems are corrupted, you could potentially lose your configuration. Save and back up your configuration files periodically. Also, before installing or migrating to a new software configuration, back up the existing configuration files.

BEFORE YOU BEGIN

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

- You are logged in to the CLI through a Telnet, or SSH connection.

- If copying to a remote location, make sure that your device has a route to the destination. Your device and the remote destination must be in the same subnetwork if you do not have a router or default gateway to route traffic between subnets.

- Using the ping command, make sure that your device has connectivity to the destination.

- Make sure that the source configuration file is in the correct directory on the remote server.

- Make sure that the permissions on the source file are set correctly. Permissions on the file should be set to world-read.

✎

Note   Use the **dir** command to ensure that enough space is available in the destination file system. If enough space is not available, use the **delete** command to remove unneeded files.

File System	Server	File Name
bootflash	sup-active sup-standby sup-1 or module-1 sup-2 or module-2 sup-local sup-remote	User-specified
volatile	—	User-specified
system	—	running-config
tftp[1]	IPv4 address, IPv6 address, or DNS name	User-specified
ftp		
scp (secure copy)		
sftp		
core	*slot-number*	Process identifier number

1.   When downloading and uploading files, a limitation of TFTP restricts file size to 32 MB on the TFTP client and 16 MB on some TFTP servers.

DETAILED STEPS

Step	Command	Purpose
Step 1	**copy** [*source filesystem***:]** *filename* [*destination filesystem***:]** *filename*	Copies a file from the specified source location to the specified destination location.
	Example: n1000v# copy system:running-config tftp://10.10.1.1/home/configs/switch3-run.cfg	Saves a copy of the running configuration to a remote switch.
	Example: n1000v# copy bootflash:system_image bootflash://sup-2/system_image	Copies a file from bootflash in the active supervisor module to bootflash in the standby supervisor module.
	Example: n1000v# copy system:running-config bootflash:my-config	Copies a running configuration to the bootflash: file system.
	Example: n1000v# copy scp://user@10.1.7.2/system-image bootflash:system-image	Copies a system image file from the SCP server identified by an IPv4 address to bootflash.
	Example: n1000v# copy sftp://172.16.10.100/myscript.txt volatile:myscript.txt	Copies a script file from the SFTP server identified by an IPv4 address to the volatile: file system.
	Example: n1000v# copy system:running-config bootflash:my-config	Places a back up copy of the running configuration on the bootflash: file system (ASCII file).
	Example: n1000v# copy bootflash:samplefile bootflash:mystorage/samplefile	Copies the file called samplefile from the root directory of the bootflash: file system to the mystorage directory.
	Example: n1000v# copy samplefile mystorage/samplefile	Copies a file within the current file system.
	Example: n1000v# copy tftp://10.10.1.1/home/configs/switch3-run.cfg system:running-config	Copies the source file to the running configuration on the switch, and configures the switch as the file is parsed line by line.

Creating a Directory

Use this procedure to create a directory at the current directory level or at a specified directory level.

Step	Command	Purpose
Step 1	**mkdir** *directory name* **dir** *filename*	Creates a directory at the current directory level
	example: n1000v# mkdir bootflash:test n1000v#	Creates a directory called test in the bootflash: directory.
	example: n1000v# mkdir test n1000v#	Creates a directory called test at the current directory level. If the current directory is bootflash:mydir, this command creates a directory called bootflash:mydir/test.

Removing an Existing Directory

Use this section to remove an existing directory from the Flash file system.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI.

- This command is only valid on Flash file systems.

- Before you can remove it, the directory must be empty.

DETAILED STEPS

Step	Command	Purpose
Step 1	**rmdir** {bootflash: | debug: | volatile:} *directory*	Removes a directory.
	example: n1000v# rmdir bootflash:test n1000v#	Removes the directory called test in the bootflash directory.
	example: n1000v# rmdir test n1000v#	Removes the directory called test at the current directory level. If the current directory is bootflash:mydir, this command deletes the bootflash:mydir/test directory.

Moving Files

Use this procedure to move a file from one location to another location.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI.

- The copy will not complete if there is not enough space in the destination directory.

⚠️

Caution   If a file with the same name already exists in the destination directory, that file is overwritten by the moved file.

DETAILED STEPS

Step	Command	Purpose
Step 1	**move** {*source path and filename*} {*destination path and filename*}	Deletes a directory.
	Example: n1000v# move bootflash:samplefile bootflash:mystorage/samplefile	Moves the file from one directory to another in the same file system (bootflash:).
	Example: n1000v# move samplefile mystorage/samplefile	Moves the file from one directory to another in the current file system.

Deleting Files or Directories

Use this procedure to delete files or directories on a Flash Memory device.

BEFORE YOU BEGIN

⚠
Caution   When deleting, if you specify a directory name instead of a file name, the entire directory and its contents are deleted.

- When you delete a file, the software erases the file.

- If you attempt to delete the configuration file or image specified by the CONFIG_FILE or BOOTLDR environment variable, the system prompts you to confirm the deletion.

- If you attempt to delete the last valid system image specified in the BOOT environment variable, the system prompts you to confirm the deletion.

DETAILED STEPS

Step	Command	Purpose
Step 1	**delete** [**bootflash:** | **debug:** | **log:** | **volatile:**] *filename or directory name* **Example:** n1000v# delete bootflash:dns_config.cfg	Deletes a specified file or directory.
	Example: n1000v# delete dns_config.cfg	Deletes the named file from the current working directory.
	Example: n1000v# delete bootflash:my-dir	Deletes the named directory and its contents.

Compressing Files

Use this procedure to compress (zip) a specified file using LZ77 coding.

BEFORE YOU BEGIN

- You are logged in to the CLI.

DETAILED STEPS

Step	Command	Purpose

Step 1	**show** `command > [`*`path`*`]` *`filename`*
	
	Example:
	`n1000v# show system internal l2fm`
	`event-history errors`
	`n1000v#`
		Directs show command output to a file.

Step 2	`dir`
	Example:
	`n1000v# dir`
		Displays the contents of the current directory, including the new file created in the first step.

Step 3	**gzip** `[`*`path`*`]` *`filename`*
	
	Example:
	`n1000v# gzip bootflash:errorsfile`
	`n1000v#`
		Compresses the specified file

Step 4	`dir`
	Example:
	`n1000v# dir`
		Displays the contents of the specified directory, including the newly-compressed file. Shows the difference in the file size of the newly-compressed file.

```
Example:
n1000v# show system internal l2fm event-history errors >errorsfile
n1000v# dir
       2687     Jul 01 18:17:20 2008  errorsfile
      16384     Jun 30 05:17:51 2008  lost+found/
       4096     Jun 30 05:18:29 2008  routing-sw/
         49     Jul 01 17:09:18 2008  sample_test.txt
    1322843     Jun 30 05:17:56 2008  nexus-1000v-dplug-mzg.4.0.4.SV1.0.42.bin
   21629952     Jun 30 05:18:02 2008  nexus-1000v-kickstart-mzg.4.0.4.SV1.0.42.bin
   39289400     Jun 30 05:18:14 2008  nexus-1000v-mzg.4.0.4.SV1.0.42.bin

Usage for bootflash://
  258408448 bytes used
 2939531264 bytes free
 3197939712 bytes total
n1000v# gzip bootflash:errorsfile
n1000v# dir
       1681     Jun 30 05:21:08 2008  cisco_svs_certificate.pem
        703     Jul 01 18:17:20 2008  errorsfile.gz
      16384     Jun 30 05:17:51 2008  lost+found/
       4096     Jun 30 05:18:29 2008  routing-sw/
         49     Jul 01 17:09:18 2008  sample_test.txt
    1322843     Jun 30 05:17:56 2008  nexus-1000v-dplug-mzg.4.0.4.SV1.0.42.bin
   21629952     Jun 30 05:18:02 2008  nexus-1000v-kickstart-mzg.4.0.4.SV1.0.42.bin
   39289400     Jun 30 05:18:14 2008  nexus-1000v-mzg.4.0.0.S1.0.34.bin
```

```
                Usage for bootflash://
                  258408448 bytes used
                 2939531264 bytes free
                 3197939712 bytes total
                n1000v#
```

Uncompressing Files

Use this procedure to uncompress (unzip) a specified file that is compressed using LZ77 coding.

BEFORE YOU BEGIN

- You are logged in to the CLI.

DETAILED STEPS

Step	Command	Purpose
Step 1	gunzip [path] *filename*	Uncompresses the specified file.
Step 2	dir	Displays the contents of a directory, including the newly uncompressed file.

```
                Example:
                n1000v# gunzip bootflash:errorsfile.gz
                n1000v# dir bootflash:
                      2687      Jul 01 18:17:20 2008   errorsfile
                     16384      Jun 30 05:17:51 2008   lost+found/
                      4096      Jun 30 05:18:29 2008   routing-sw/
                        49      Jul 01 17:09:18 2008   sample_test.txt
                   1322843      Jun 30 05:17:56 2008   nexus-1000v-dplug-mzg.4.0.0.SV1.0.42.bin
                  21629952      Jun 30 05:18:02 2008   nexus-1000v-kickstart-mzg.4.0.4.SV1.0.42.bin
                  39289400      Jun 30 05:18:14 2008   nexus-1000v-mzg.4.0.0.SV1.0424.bin

                Usage for bootflash://sup-local
                  258408448 bytes used
                 2939531264 bytes free
                 3197939712 bytes total
                DCOS-112-R5#
```

Directing Command Output to a File

Use this procedure to direct command output to a file.

DETAILED STEPS

Step	Command	Purpose
Step 1	`show running-config >` [*path* | *filename*]	Directs the output of the command, **show running-config,** to a path and filename.

	Example:
	`n1000v# show running-config >`
	`volatile:switch1-run.cfg`
		Directs the output of the command, **show running-config,** to the file, switch1-run.cfg, on the volatile file system.

	Example:
	`n1000v# show running-config >`
	`bootflash:switch2-run.cfg`
		Directs the output of the command, **show running-config,** to the file, switch2-run.cfg, in bootflash.

	Example:
	`n1000v# show running-config >`
	`tftp://10.10.1.1/home/configs/switch3-run.cfg`
		Directs the output of the command, **show running-config,** to the file, switch3-run.cfg, on a TFTP server.

	Example: `n1000v# show interface > samplefile`	Directs the output of the command, show interface, to the file, samplefile, at the same directory level, for example, in bootflash.

Verifying a Configuration File before Loading

Use this procedure to verify the integrity of an image before loading it. This command can be used for both the system and kickstart images.

DETAILED STEPS

Step	Command	Purpose

Step 1	`copy` *source path and file* `system:running-config`
	
	Example:
	`n1000v# copy`
	`tftp://10.10.1.1/home/configs/switch3-run.cfg`
	`system:running-config`
		Copies the source file to the running configuration on the switch, and configures the switch as the file is parsed line by line.

Step 2	`show version image` [bootflash: | modflash: |volatile:]
	
	Example:
	`n1000v# show version image bootflash:isan.bin`
	`  image name: nexus-1000v-mz.4.0.4.SV1.1.bin`
	`  bios:      version unavailable`
	`  system:    version 4.0(4)SV1(1)`
	`  compiled:  4/2/2009 23:00:00 [04/23/2009 09:55:29]`
	`n1000v#`
		Validates the specified image.

Rolling Back to a Previous Configuration

Use this procedure to recover your configuration from a previously saved version.

BEFORE YOU BEGIN

Note Each time a **copy running-config startup-config** command is used, a binary file is created and the ASCII file is updated. A valid binary configuration file reduces the overall boot time significantly. A binary file cannot be uploaded, but its contents can be used to overwrite the existing startup configuration. The **write erase** command clears the binary file.

DETAILED STEPS

Step	Command	Purpose
Step 1	copy running-config bootflash: {*filename*} **Example:** n1000v# copy running-config bootflash:June03-Running	Reverts to a snapshot copy of a previously saved running configuration (binary file).
	copy bootflash: {*filename***} startup-config** **Example:** n1000v# copy bootflash:my-config startup-config	Reverts to a configuration copy that was previously saved in the bootflash: file system (ASCII file).

Displaying Files

This section describes how to display information about files and includes the following procedures:

Displaying File Contents

Use this procedure to display the contents of a specified file.

BEFORE YOU BEGIN

- You are logged in to the CLI.

DETAILED STEPS

Step	Command	Purpose
Step 1	`show file [bootflash:	debug:	volatile:]` *filename*  **Example:** `n1000v# show file bootflash:sample_test.txt` `config t` `Int veth1/1` `no shut` `end` `show int veth1/1`  `n1000v#`	Displays the contents of the specified file.

Displaying Directory Contents

Use this procedure to display the contents of a directory or file system.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI.

Step	Command	Purpose

Step 1	`pwd`
	
	Example:
	`n1000v# pwd`
	`bootflash:`
		Displays the present working directory.

Step 2	`dir`	Displays the contents of the directory.

```
Example:
n1000v# pwd
bootflash:
n1000v# dir

Usage for volatile://
         0 bytes used
  20971520 bytes free
  20971520 bytes total
n1000v#
```

Displaying File Checksums

Use this procedure to display checksums for checking file integrity.

Step	Command	Purpose
Step 1	**show file** *filename* [**cksum	md5sum**] **Example:** n1000v# **s**how file bootflash:cisco_svs_certificate.pem cksum 266988670	Provides the checksum or MD5 checksum of the file for comparison with the original file.
	Example: n1000v# show file bootflash:cisco_svs_certificate.pem md5sum d3013f73aea3fda329f7ea5851ae81ff n1000v#	Provides the Message-Digest Algorithm 5 (MD5) checksum of the file. MD5 is an electronic fingerprint for the file.

Displaying the Last Lines in a File

Use this command to display the last lines (tail end) of a specified file.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.

DETAILED STEPS

Step	Command	Purpose
Step 1	**tail** {*path*}[*filename*] {*Number of lines*}	Displays the requested number of lines from the end of the specified file. Allowable range for number of lines: 0 - 80

```
Example:
n1000v# tail bootflash:errorsfile 5

20) Event:E_DEBUG, length:34, at 171590 usecs after Tue Jul  1 09:29:05 2008
    [102] main(326): stateless restart


n1000v#
```

Feature History for File Management

This section provides the file management feature release history.

Feature Name	Releases	Feature Information
File Management	4.0(4)SV1(1)	This feature was introduced.

Managing Users

Information About User Management

You can identify the users currently connected to the device and send a message to either a single user or all users.

For information about assigning user roles, see the *Cisco Nexus 1000V Security Configuration Guide, Release 4.0(4)SV1(3)*.

Displaying Current User Access

Use this procedure to display all users currently accessing the switch.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI.

DETAILED STEPS

	Command	Description
Step 1	`show users`	Displays a list of users who are currently accessing the system.

```
Example:
n1000v# show users
NAME     LINE         TIME          IDLE        PID COMMENT
admin    pts/0        Jul  1 04:40 03:29        2915 (::ffff:64.103.145.136)
admin    pts/2        Jul  1 10:06 03:37        6413 (::ffff:64.103.145.136)
admin    pts/3        Jul  1 13:49   .          8835 (171.71.55.196)*
n1000v#
```

Sending a Message to Users

Use this command to send a message to all active CLI users currently using the system.

BEFORE YOU BEGIN

Before using this command, you must know or do the following:

- You are logged in to the CLI.

DETAILED STEPS

	Command	Description
Step 1	`send {session device} line`	Sends a message to users currently logged in to the system.
		- **session**: sends the message to a specified pts/tty device type.
		- *line*: a message of up to 80 alphanumeric characters in length.

```
Example:
n1000v# send Hello. Shutting down the system in 10 minutes.

Broadcast Message from admin@switch
        (/dev/pts/34) at 8:58 ...

Hello. Shutting down the system in 10 minutes.


n1000v#
```

Feature History for User Management

This section provides the user management feature release history.

Feature Name	Releases	Feature Information
User Management	4.0(4)SV1(1)	This feature was introduced.

C H A P T E R **8**

Configuring NTP

This chapter describes how to configure the Network Time Protocol (NTP).

Information about NTP

The Network Time Protocol (NTP) synchronizes timekeeping among a set of distributed time servers and clients. This synchronization allows you to correlate events when you receive system logs and other time-specific events from multiple network devices.

NTP uses the User Datagram Protocol (UDP) as its transport protocol. All NTP communication uses the Universal Time Coordinated (UTC) standard. An NTP server usually receives its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to within a millisecond of each other.

NTP uses a stratum to describe how many NTP hops away that a network device is from an authoritative time source. A stratum 1 time server has an authoritative time source (such as an atomic clock) directly attached to the server. A stratum 2 NTP server receives its time through NTP from a stratum 1 NTP server, which in turn connects to the authoritative time source.

NTP avoids synchronizing to a network device that may keep accurate time. NTP never synchronizes to a system that is not in turn synchronized itself. NTP compares the time reported by several network devices and does not synchronize to a network device that has a time that is significantly different than the others, even if its stratum is lower.

Cisco NX-OS cannot act as a stratum 1 server. You cannot connect to a radio or atomic clock. We recommend that the time service that you use for your network is derived from the public NTP servers available on the Internet.

If the network is isolated from the Internet, Cisco NX-OS allows you to configure a network device so that the device acts as though it is synchronized through NTP, when in fact it has determined the time using other means. Other network devices can then synchronize to that network device through NTP.

NTP Peers

NTP allows you to create a peer relationship between two networking devices. A peer can provide time on its own or connect to an NTP server. If both the local device and the remote peer point to different NTP servers, your NTP service is more reliable. The local device maintains the right time even if its NTP server fails by using the time from the peer.

Figure 8-1 displays a network with two NTP stratum 2 servers and two switches.

Figure 8-1   NTP Peer and Server Association
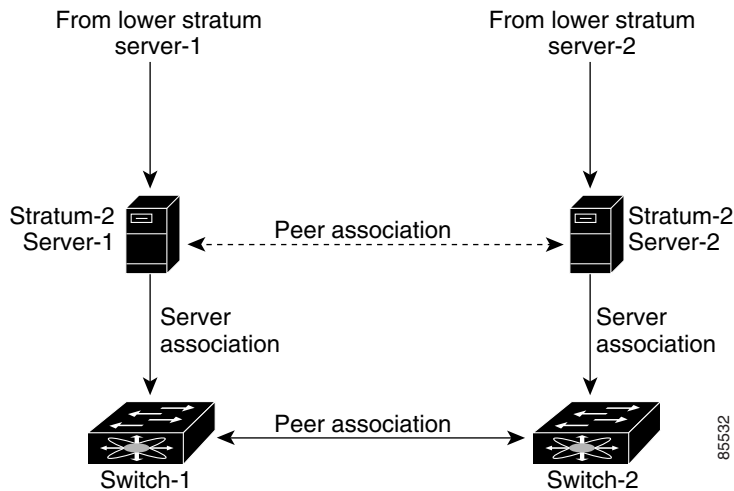
In this configuration, switch 1 and switch 2 are NTP peers. switch 1 uses stratum-2 server 1, while switch 2 uses stratum-2 server 2. If stratum-2 server-1 fails, switch 1 maintains the correct time through its peer association with switch 2.

High Availability

Stateless restarts are supported for NTP. After a reboot or a supervisor switchover, the running configuration is applied.

You can configure NTP peers to provide redundancy in case an NTP server fails.

Prerequisites for NTP

If you configure NTP, you must have connectivity to at least one server that is running NTP.

Configuration Guidelines and Limitations

NTP has the following configuration guidelines and limitations:

- You should have a peer association with another device only when you are sure that your clock is reliable (which means that you are a client of a reliable NTP server).
- A peer configured alone takes on the role of a server and should be used as backup. If you have two servers, you can configure several devices to point to one server and the remaining devices point to the other server. You can then configure peer association between these two servers to create a more reliable NTP configuration.
- If you only have one server, you should configure all the devices as clients to that server.
- You can configure up to 64 NTP entities (servers and peers).

Default Settings

The following table lists the default settings for CDP and NTP parameters.

Parameter	Default
NTP	Enabled

Configuring an NTP Server and Peer

Use this procedure to configure an NTP server and peer.

BEFORE YOU BEGIN

- You can configure NTP using IPv4 addresses or domain name server (DNS) names.

SUMMARY STEPS

1. **config t**
2. **ntp server** {*ip-address* | *ipv6-address* | *dns-name*}
3. **ntp peer** {*ip-address* | *ipv6-address* | *dns-name*}
4. **show ntp peers**
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into the CLI Global Configuration mode.

Step 2	`ntp server {ip-address	ipv6-address	dns-name}`  **Example:** `n1000v(config)# ntp server 192.0.2.10`	Forms an association with a server.
Step 3	`ntp peer {ip-address	dns-name}`  `n1000v(config)# ntp peer 2001:0db8::4101`	Forms an association with a peer. You can specify multiple peer associations.

Step 4	`show ntp peers`
	
	Example:
	`n1000v(config)# show ntp peers`
		(Optional) Displays the configured server and peers.
		
		Note A domain name is resolved only when you have a DNS server configured.

Step 5	`copy running-config startup-config`
	
	Example:
	`n1000v(config-if)# copy running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

The following is an example configures an NTP server and peer:

```
n1000v# config t
n1000v(config)# ntp server 192.0.2.10
n1000v(config# ntp peer 2001:0db8::4101
```

Clearing NTP Statistics or Sessions

Use the following commands to clear NTP statistics or sessions.

Command	Purpose
clear ntp statistics	Clears the NTP statistics.
clear ntp session	Clears the NTP sessions.

Verifying the NTP Configuration

To display NTP configuration information, use one of the following commands:

Command	Purpose
show ntp peer-status	Displays the status for all NTP servers and peers.
show ntp peers	Displays all the NTP peers.
show ntp statistics {io | local | memory | peer {ip-address | dns-name}	Displays the NTP statistics

NTP Example Configuration

This example configures an NTP server:

```
config t
ntp server 192.0.2.10
```

Additional References

For additional information related to NTP, see the following sections:

- Related Documents, page 8-5
- Standards, page 8-5

Related Documents

Related Topic	Document Title
Interface	*Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(3)*
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(3)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for NTP

This section shows the NTP feature release history.

Feature Name	Releases	Feature Information
NTP	4.0(4)SV1(1)	This feature was introduced.

Configuring Local SPAN and ERSPAN

This chapter describes how to configure the Local and ER Ethernet switched port analyzer (SPAN) feature to monitor traffic.

Information About SPAN and ERSPAN

The Switched Port Analyzer (SPAN) feature (sometimes called port mirroring or port monitoring) allows network traffic to be analyzed by a network analyzer such as a Cisco SwitchProbe or other Remote Monitoring (RMON) probe.

SPAN lets you monitor traffic on one or more ports, or one or more VLANs, and send the monitored traffic to one or more destination ports where the network analyzer is attached.

SPAN Sources

The interfaces from which traffic can be monitored are called SPAN sources. These include Ethernet, virtual Ethernet, port-channel, and VLAN. When a VLAN is specified as a SPAN source, all supported interfaces in the VLAN are SPAN sources. Traffic can be monitored in the receive direction, the transmit direction, or both directions for Ethernet and virtual Ethernet source interfaces.

- Receive source (Rx)—Traffic that enters the switch through this source port is copied to the SPAN destination port.
- Transmit source (Tx)—Traffic that exits the switch through this source port is copied to the SPAN destination port.

Characteristics of Source Ports

A source port has these characteristics:

- Can be port type Ethernet, virtual Ethernet, port-channel, or VLAN.
- Cannot be a destination port.
- Can be configured to monitor the direction of traffic —receive, transmit, or both.
- Source ports can be in the same or different VLANs.
- For VLAN SPAN sources, all active ports in the source VLAN are included as source ports.
- Local SPAN sources must be on the same host (linecard) as the destination port.

SPAN Destinations

SPAN destinations refer to the interfaces that monitor source ports. This section includes the following topics:

- Characteristics of Local SPAN Destination Ports, page 9-2
- Characteristics of ERSPAN Destination Ports, page 9-2

Characteristics of Local SPAN Destination Ports

Each local SPAN session must have at least one destination port (also called a monitoring port) that receives a copy of traffic from the source ports or VLANs. A destination port has these characteristics:

- Can be any physical or virtual Ethernet port or a port channel.
- Cannot be a source port.
- Is excluded from the source list and is not monitored if it belongs to a source VLAN of any SPAN session.
- Receives copies of transmitted and received traffic for all monitored source ports. If a destination port is oversubscribed, it can become congested. This congestion can affect traffic forwarding on one or more of the source ports.
- Must be on the same host (linecard) as the source port.
- In Local SPAN, the source interface and destination interface are on the same device. See Figure 9-1, Local SPAN.

Characteristics of ERSPAN Destination Ports

- An ERSPAN destination is specified by an IP address.
- In ERSPAN, source SPAN interface and destination SPAN interface may be on different devices interconnected by an IP network. ERSPAN traffic is GRE-encapsulated. See Figure 9-2, ERSPAN.
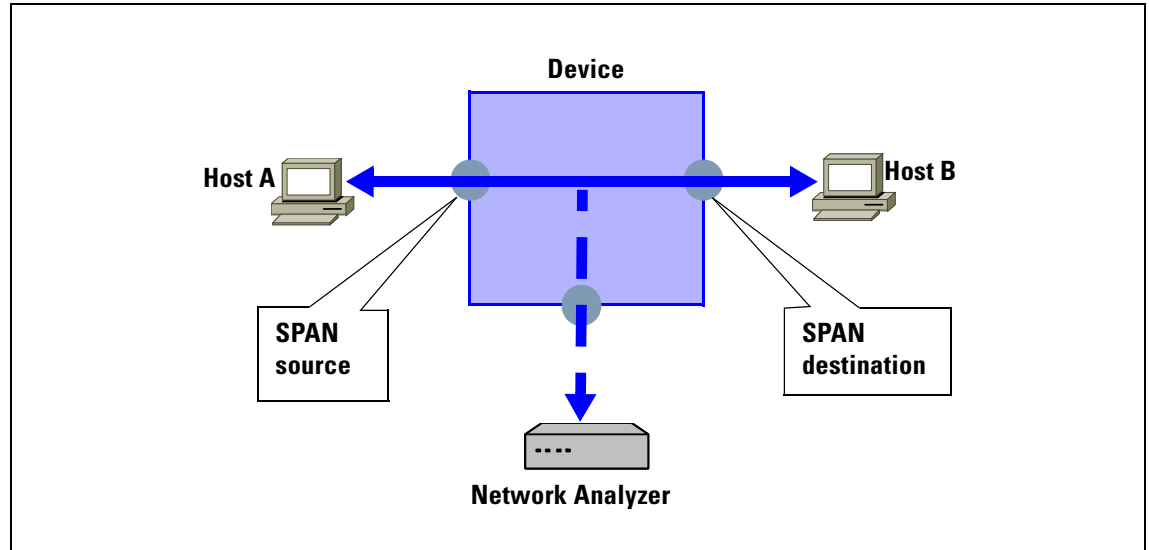
Local SPAN

In Local SPAN, the source interface and destination interface are on the same device. The network analyzer is attached directly to the SPAN destination port. The SPAN source can be a port or a VLAN interface.The destination, although usually a port, could also be a VLAN.

Figure 9-1 shows that traffic transmitted by host A is received on the SPAN source interface. Traffic (ACLs, QoS, and so forth) is processed as usual. Traffic is then replicated. The original packet is forwarded on toward host B. The replicated packet is then sent to the destination SPAN interface where the monitor is attached.

Local SPAN can replicate to one or more destination ports. Traffic can be filtered so that only traffic of interest is sent out the destination SPAN interface.

Local SPAN can monitor all traffic received on the source interface including BPDUs.

Figure 9-1 *Local SPAN*

Encapsulated Remote SPAN

Encapsulated remote (ER) SPAN monitors traffic in multiple network devices across an IP network and sends that traffic in an encapsulated envelope to destination analyzers. In contrast, Local SPAN cannot forward traffic through the IP network. ERSPAN can be used to monitor traffic remotely. ERSPAN sources can be ports or VLANs.

In Figure 9-2, the ingress and egress traffic for host A are monitored using ERSPAN. Encapsulated ERSPAN packets are routed from host A through the routed network to the destination device where they are de-capsulated and forwarded to the attached network analyzer. The destination may also be on the same L2 network as the source.

Figure 9-2   *ERSPAN*
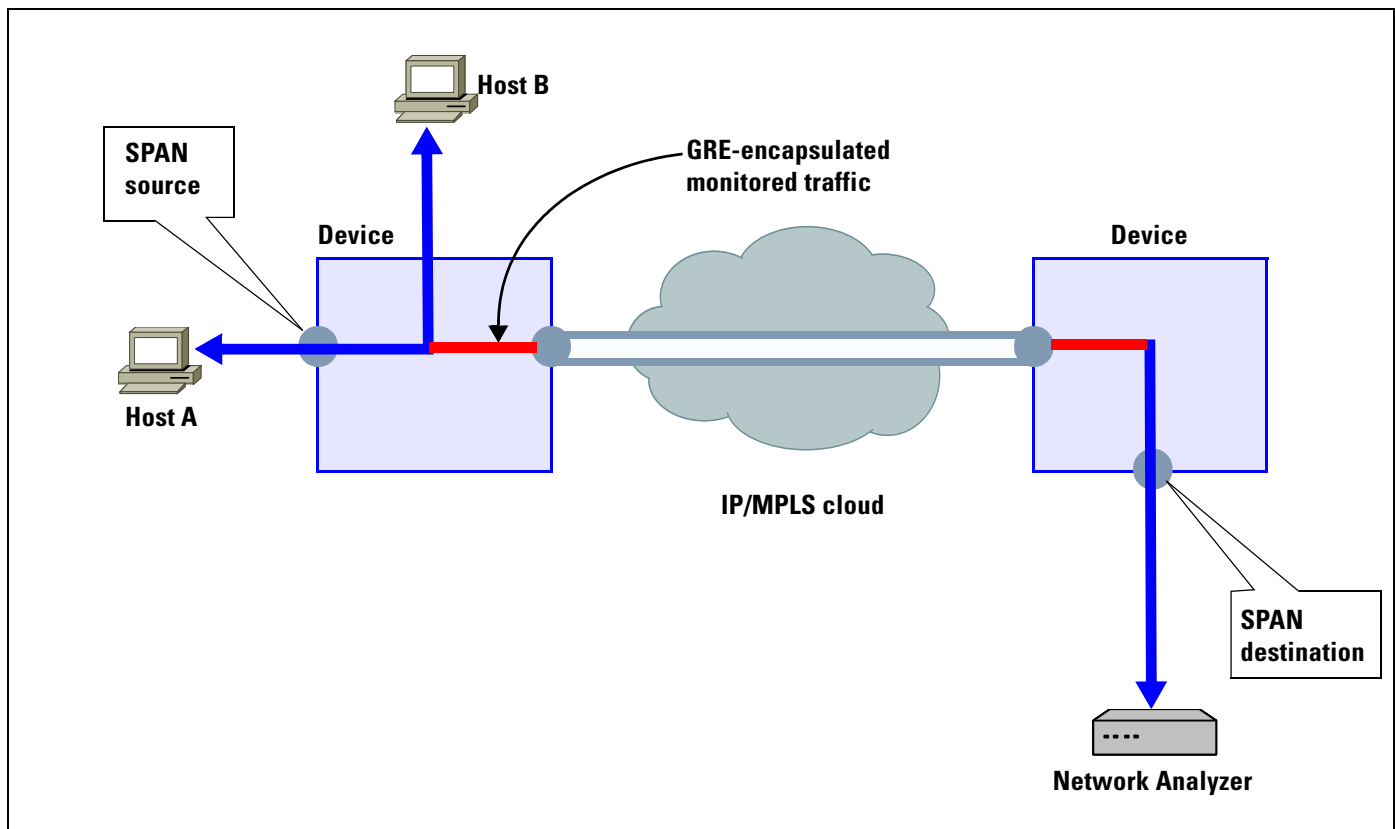
Network Analysis Module

You can also use the Cisco Network Analysis Module (NAM) to monitor ERSPAN data sources for application performance, traffic analysis, and packet header analysis.

To use NAM for monitoring the Cisco Nexus 1000V ERSPAN data sources see the *Cisco Nexus 1010 Network Analysis Module Installation and Configuration Note, 4.2*.

SPAN Sessions

You can create up to 64 total SPAN sessions (Local SPAN plus ERSPAN) on your local device.

You can configure an ERSPAN session ID that is added to the ERSPAN header of the encapsulated frame to differentiate between various ERSPAN streams of traffic at the termination box. You can also configure the range of flow ID numbers. For more information, see

You can also create a SPAN session to monitor multiple VLAN sources and choose only VLANs of interest to transmit on multiple destination ports. For example, you can configure SPAN on a trunk port and monitor traffic from different VLANs on different destination ports.
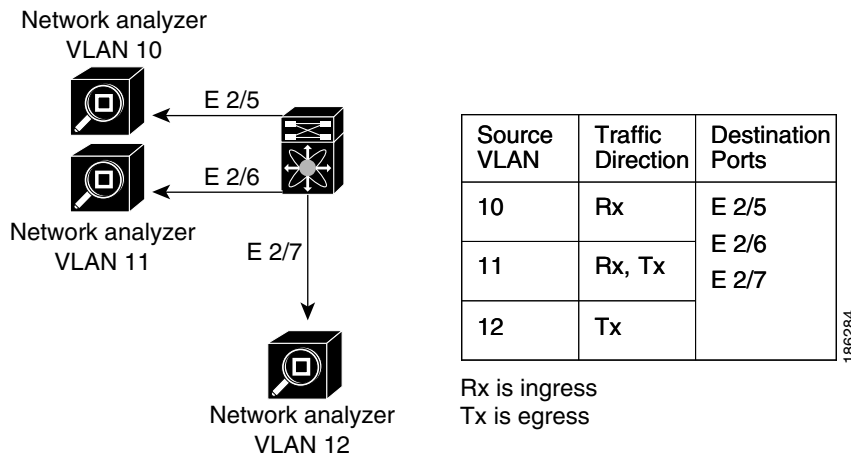
Figure 9-3 shows a VLAN-based SPAN configuration that copies traffic from three VLANs to three specified destination ports. You can choose which VLANs to allow on each destination port to limit the traffic transmitted. In Figure 9-3, the device transmits packets from one VLAN at each destination port.

Note   VLAN-based SPAN sessions cause all source packets to be copied to all destinations, whether the packets are required at the destination or not. VLAN traffic filtering occurs at transmit destination ports.

Figure 9-3   VLAN-based SPAN Configuration

Source VLAN	Traffic Direction	Destination Ports
10	Rx	E 2/5
11	Rx, Tx	E 2/6
		E 2/7
12	Tx	

Rx is ingress
Tx is egress

SPAN Guidelines and Limitations

SPAN has the following configuration guidelines and limitations:

- A maximum of 64 SPAN sessions (Local SPAN plus ERSPAN) can be configured on the VSM.
- A maximum of 32 source VLANs are allowed in a session.
- A maximum of 128 source interfaces are allowed in a session.

Caution   **Overload Potential**
To avoid an overload on uplink ports, use caution when configuring ERSPAN, especially when sourcing VLANs.

- A port can be configured in a maximum of 4 SPAN sessions.
- The destination port used in one SPAN session cannot also be used as the destination port for another SPAN session.
- You cannot configure a port as both a source and destination port.

- In a SPAN session, packets that source ports receive may be replicated even though they are not transmitted on the ports. The following are examples of this behavior:
  - Traffic that results from flooding
  - Broadcast and multicast traffic
- For VLAN SPAN sessions switched on the same VLAN with both receive and transmit configured, two packets (one from receive and one from transmit) are forwarded from the destination port.

Default Settings

Table 9-1 lists the SPAN default settings.

Table 9-1   SPAN Defaults

Parameters	Default
State	SPAN sessions are created in the shut state.
Description	blank
Traffic direction for the source interface	both

Configuring SPAN

This section describes how to configure SPAN and includes the following procedures.

Configuring a Local SPAN Session

Use this procedure to configure a SPAN session.

Note   If you are configuring ERSPAN, see the "Configuring an ERSPAN Session" procedure on page 9-13.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.
- You know the number of the SPAN session you are going to configure.
- The source and destination ports are already configured in either access or trunk mode. For more information, see the *Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(3)*.
- SPAN sessions are created in the shut state by default.

- When you create a SPAN session that already exists, any additional configuration is added to that session. To make sure the session is cleared of any previous configuration, you can delete the session first (see Step 2, **no monitor session**).

- This procedure involves creating the SPAN session in Monitor Configuration mode; and then, optionally, configuring allowed VLANs in Interface Configuration mode.

SUMMARY STEPS

1. **config t**

2. **no monitor session** *session-number*

3. **monitor session** *session-number*

4. **description** *description*

5. **source** {**interface** *type* | **vlan**} {*number* | *range*} [**rx** | **tx** | **both**]

6. (Optional) Repeat Step 5 to configure additional SPAN sources.

7. (Optional) **filter vlan** {*number* | *range*}

8. (Optional) Repeat Step 7 to configure all source VLANs to filter.

9. **destination interface** *type* {*number* | *range*}

10. (Optional) Repeat Step 9 to configure all SPAN destination ports.

11. **no shut**

12. (Optional) **exit**

13. (Optional) **interface ethernet** *slot*/*port*[*-port*]

14. (Optional) **switchport trunk allowed vlan** {*vlan-range* | **add** *vlan-range* | **except** *vlan-range* | **remove** *vlan-range* | **all** | **none**}

15. (Optional) Repeat Step 13 and Step 14 to configure the allowed VLANs on each destination port.

16. (Optional) **show interface ethernet** *slot*/*port*[*-port*] **trunk**

17. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**config t** Example: n1000v# config t n1000v(config)#	Places you in the CLI Global Configuration mode.
Step 2	**no monitor session** *session-number* Example: n1000v(config)# no monitor session 3	Clears the specified session.
Step 3	**monitor session** *session-number* Example: n1000v(config)# monitor session 3 n1000v(config-monitor)#	Creates a session with the given session number and places you in the CLI Monitor Configuration mode to further configure the session.

	Command	Purpose
Step 4	**description** *description* **Example:** n1000v(config-monitor)# description my_span_session_3	For the specified SPAN session, adds a description. • description: up to 32 alphanumeric characters default = blank (no description)
Step 5	**source** {**interface** *type* | **vlan**} {*number* | *range*} [**rx** | **tx** | **both**] **Example 1:** n1000v(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx **Example 2:** n1000v(config-monitor)# source interface port-channel 2 **Example 3:** n1000v(config-monitor)# source interface vethernet 12 both **Example 4:** n1000v(config-monitor)# source vlan 3, 6-8 tx	For the specified session, configures the sources and the direction of traffic to monitor. • type: Specify the interface type—ethernet, port-channel, vethernet. • number: Specify the interface slot/port or range; or the VLAN number or range to monitor. • traffic direction: Specify traffic monitoring to be in one of the following directions: – receive (rx) (the VLAN default) – transmit (tx) – both (the interface default)
Step 6	(Optional) Repeat Step 5 to configure additional SPAN sources.	
Step 7	**filter vlan** {*number* | *range*} **Example:** n1000v(config-monitor)# filter vlan 3-5, 7	(Optional) For the specified SPAN session, configures the filter from among the source VLANs.
Step 8	(Optional) Repeat Step 7 to configure all source VLANs to filter.	
Step 9	**destination interface** *type* {*number* | *range*} **Example:** n1000v(config-monitor)# destination interface ethernet 2/5, ethernet 3/7	For the specified SPAN session, configures the port(s) to act as destination(s) for copied source packets. • type: Specify the interface type—ethernet or vethernet. • number: Specify the interface slot/port to monitor. • range: Specify the interface range to monitor. **Note** SPAN destination ports must already be configured as either access or trunk ports.
Step 10	(Optional) Repeat Step 9 to configure all SPAN destination ports.	
Step 11	no shut **Example:** n1000v(config-monitor)# no shut	Enables the SPAN session. By default, the session is created in the shut state.
Step 12	**exit** **Example:** n1000v(config-monitor)# exit n1000v(config)#	(Optional) Exits Monitor Configuration mode and places you in CLI Configuration mode.

	Command	Purpose
Step 13	**interface ethernet** *slot*/*port*[*-port*] **Example:** n1000v(config)# interface ethernet 2/5 n1000v(config-if)#	(Optional) Places you in CLI Interface Configuration mode for the specified interface.
Step 14	**switchport trunk allowed vlan** {*vlan-range* | **add** *vlan-range* | **except** *vlan-range* | **remove** *vlan-range* | **all** | **none**} **Example:** n1000v(config-if)# switchport trunk allowed vlan 3-5	(Optional) For the specified interface. configures the range of VLANS that are allowed on the interface. By default all VLANs are allowed on the interface. • **vlan-range**: Specify a range of VLANs to be allowed on the interface. • **add vlan-range**: Add to the existing VLANs allowed on the interface. • **except vlan-range**: Exclude a range of VLANs from those allowed on the interface • **remove vlan-range**: Remove a range of VLANs from those allowed on the interface. • **all**: Allow all VLANs on the interface. This is the default. • **none**: Allow no VLANs on the interface.
Step 15	(Optional) Repeat Step 13 and Step 14 to configure the allowed VLANs on each destination port.	
Step 16	**show interface ethernet** *slot*/*port*[*-port*] **trunk** **Example:** n1000v(config-if)# show interface ethernet 2/5 trunk	(Optional) Displays the interface trunking configuration for the selected slot and port or range of ports.
Step 17	**copy running-config startup-config** **Example:** n1000v(config-if)# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Configuring an ERSPAN Port Profile

Use this procedure to configure a port profile on the VSM to carry ERSPAN packets through the IP network to a remote destination analyzer.

BEFORE YOU BEGIN

• You are logged in to the CLI in EXEC mode.

• This configuration must be completed for all hosts in the vCenter Server.

• You know the name to be used for this port profile.

Note   The port profile name is used to configure the VMKNIC that is required on each of the ESX hosts.

- You know the name of the VMware port group to which this profile maps.

- You have the VMware documentation for adding a new virtual adapter.

- You have already created the system VLAN and you know its VLAN ID which will be used in this configuration.

  For more information about system port profiles, see the *Cisco Nexus 1000V Port Profile Configuration Guide, Release 4.0(4)SV1(3)*.

- The port profile used for ERSPAN must be configured for Layer 3 control. This procedure includes a step for making this configuration.

- Only one VM kernel NIC can be assigned to this Layer 3 control port profile per host.

  – If more than one VMware kernel NIC is assigned to a host, the last one assigned takes effect.

  – If more than one VMware kernel NIC is assigned to a host, and you remove the second one assigned, then the VEM does not use the first one assigned. Instead, you must remove both VMware kernel NICs and then add one back.

- The port profile must be an access port profile. It cannot be a trunk port profile. This procedure includes steps to configure the port profile as an access port profile.

- For more information about creating a Layer 3 control port profile, see the "Creating a Port Profile for Layer 3 Control" procedure on page 3-9.

SUMMARY STEPS

1. **config t**
2. **port-profile** *port_profile_name*
3. **capability l3control**
4. **vmware port-group** *pg_name*
5. **switchport mode access**
6. **switchport access vlan** *vlan_id*
7. **no shutdown**
8. **system vlan** *vlan_id*
9. **state enabled**
10. (Optional) **show port-profile name** *port_profile_name*
11. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	`port-profile` *port_profile_name*
	
	Example:
	`n1000v(config)# port-profile`
	`erspan_profile`
	`n1000v(config-port-prof)#`
		Creates the port profile and places you into CLI Global Configuration mode for the specified port profile. Saves the port profile in the running configuration.
		
		The port-profile name can be up to 80 characters and must be unique for each port profile on the Cisco Nexus 1000V.

Step 3	`capability l3control`
	
	Example:
	`n1000v(config-port-prof)# capability`
	`l3control`
	`n1000v(config-port-prof)#`
		Configures the port-profile to carry ERSPAN packet traffic through the IP network and saves this in the running configuration.

Step 4	`vmware port-group` *pg_name*
	
	Example:
	`n1000v(config-port-prof)#vmware port-group`
	`erspan`
	`n1000v(config-port-prof)#`
		Designates the port profile as a VMware port group and adds the name of the VMware port group to which this profile maps. Saves the settings in the running configuration.
		
		The port profile is mapped to a VMware port group of the same name. When a vCenter Server connection is established, the port group created in Cisco Nexus 1000V is then distributed to the virtual switch on the vCenter Server.
		
		• **pg-name**: Port group name. If you do not specify a pg-name, then the port group name will be the same as the port profile name. If you want to map the port profile to a different port group name, use the pg-name option followed by the alternate name.

Step 5	`switchport mode access`
	
	Example:
	`n1000v(config-port-prof)# switchport mode`
	`access`
	`n1000v(config-port-prof)#`
		Designates the interfaces as switch access ports (the default).

Step 6	`switchport access vlan` *vlan_id*
	
	Example 1:
	`n1000v(config-port-prof)# switchport`
	`access vlan 2`
	`n1000v(config-port-prof)#`
		Assigns a VLAN ID to the access port for this port profile and saves the setting in the running configuration.

Step 7	`no shutdown`
	
	Example:
	`n1000v(config-port-prof)# no shutdown`
	`n1000v(config-port-prof)#`
		Enables the interface in the running configuration.

	Command	Purpose
Step 8	**system vlan** *vlan_id* **Example:** n1000v(config-port-prof)# system vlan 2 n1000v(config-port-prof)#	Associates the system VLAN ID with the port profile and saves it in the running configuration. Must match the VLAN ID assigned to the access port. If it does not match, then the following error message is generated: ERROR: System vlan being set does not match the switchport access vlan 2
Step 9	**state enabled** **Example:** n1000v(config-port-prof)# state enabled n1000v(config-port-prof)#	Enables the port profile in the running configuration. This port profile is now ready to send out ERSPAN packets on all ESX Hosts with ERSPAN sources

Step 10	**show port-profile name** *port_profile_name*
	
	Example:
	n1000v(config-port-prof)# show
	port-profile name erspan
	port-profile erspan
	  description:
	  status: enabled
	  capability uplink: no
	  capability l3control: yes
	  system vlans: 2
	  port-group: access
	  max-ports: 32
	  inherit:
	  config attributes:
	    switchport access vlan 2
	    no shutdown
	  evaluated config attributes:
	    switchport access vlan 2
	    no shutdown
	  assigned interfaces:
	
	n1000v(config-port-prof)#
		(Optional) Displays the configuration for the specified port profile as it exists in the running configuration.

Step 11	**copy running-config startup-config** **Example:** n1000v(config-port-prof)# copy running-config startup-config [##] 100% n1000v(config-port-prof)#	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.
Step 12	Using the VMware documentation, go to vSphere Client and configure a VMKNIC on each ESX Host. Make sure the VMKNIC points to this port profile as a **new virtual adapter**.	

Configuring an ERSPAN Session

Use this procedure to configure an ERSPAN session.

Note   If you are configuring Local SPAN, see the "Configuring a Local SPAN Session" procedure on page 9-6.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.

- You know the number of the SPAN session you are going to configure.

- You have already configured an ERSPAN-capable port profile on the VSM using the "Configuring an ERSPAN Port Profile" procedure on page 9-9.

- Using the VMware documentation for adding a new virtual adapter, you have already configured the required VMKNIC on each of the ESX hosts.

- SPAN sessions are created in the shut state by default.

- When you create a SPAN session that already exists, any additional configuration is added to that session. To make sure the session is cleared of any previous configuration, you can delete the session first (see Step 2, **no monitor session**).

- This procedure involves creating the SPAN session in ERSPAN Source Configuration mode.

SUMMARY STEPS

1. **config t**

2. **no monitor session** *session-number*

3. **monitor session** *session-number* **type erspan-source**

4. **description** *description*

5. **source** {**interface** *type* | **vlan**} {*number* | *range*} [**rx** | **tx** | **both**]

6. (Optional) Repeat Step 5 to configure additional ERSPAN sources.

7. (Optional) **filter vlan** {*number* | *range*}

8. (Optional) Repeat Step 7 to configure all source VLANs to filter.

9. **destination ip** *ip_address*

10. (Optional) **ip ttl** *ttl_value*

11. (Optional) **ip prec** *ipp_value*

12. (Optional) **ip dscp** *dscp_value*

13. (Optional) **mtu** *mtu_value*

14. **erspan-id** *flow_id*

15. **no shut**

16. (Optional) **show monitor session session_id**

17. (Optional) **exit**

18. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	`no monitor session` *session-number* **Example:** `n1000v(config)# no monitor session 3`	Clears the specified session.

Step 3	`monitor session` *session-number* `type erspan-source`
	
	Example:
	`n1000v(config)# monitor session 3 type erspan`
	`n1000v(config-erspan-source)#`
		Creates a session with the given session number and places you in the CLI ERSPAN Source Configuration mode. This configuration is saved in the running configuration.

Step 4	`description` *description*
	Example:
	`n1000v(config-erspan-src)# description my_erspan_session_3`
	`n1000v(config-erspan-src)#`
		For the specified ERSPAN session, adds a description and saves it in the running configuration.
		
		• description: up to 32 alphanumeric characters default = blank (no description)

Step 5	`source` {`interface` *type* | `vlan`} {*number* | *range*} [`rx` | `tx` | `both`]
	
	Example 1:
	`n1000v(config-erspan-src)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	
	Example 2:
	`n1000v(config-erspan-src)# source interface port-channel 2`
	
	Example 3:
	`n1000v(config-erspan-src)# source interface vethernet 12 both`
	
	Example 4:
	`n1000v(config-erspan-src)# source vlan 3, 6-8 tx`
		For the specified session, configures the source(s) and the direction of traffic to monitor, and saves them in the running configuration.
		
		• type: Specify the interface type—ethernet, port-channel, vethernet.
		
		• number: Specify the interface slot/port or range; or the VLAN number or range to monitor.
		
		• traffic direction: Specify traffic monitoring to be in one of the following directions:
		 – receive (rx) (the VLAN default)
		 – transmit (tx)
		 – both (the interface default)

Step 6	(Optional) Repeat Step 5 to configure additional ERSPAN sources.	
Step 7	`filter vlan` {*number* | *range*} **Example:** `n1000v(config-erspan-src)# filter vlan 3-5, 7`	(Optional) For the specified ERSPAN session, configures the VLANs, VLAN lists, or VLAN ranges to be monitored; and saves this in the running configuration. On the monitor port, only the traffic from the VLANs which match the VLAN filter list are replicated to the destination.
Step 8	(Optional) Repeat Step 7 to configure all source VLANs to filter.	

	Command	Purpose
Step 9	**destination ip** *ip_address* **Example:** n1000v(config-erspan-src)# destination ip 10.54.54.1 n1000v(config-monitor-erspan-src)#	Configures the IP address of the host to which the encapsulated traffic is sent and saves it in the running configuration.
Step 10	**ip ttl** *ttl_value* **Example:** n1000v(config-monitor-erspan-src)# **ip ttl 64** n1000v(config-monitor-erspan-src)#	(Optional) Specifies the IP time-to-live value, from 1-255, for the packets in the ERSPAN traffic, and saves it in the running configuration.
Step 11	**ip prec** *precedence_value* **Example:** n1000v(config-monitor-erspan-src)# **ip prec 1** n1000v(config-monitor-erspan-src)#	(Optional) Specifies the IP precedence value, from 0-7, for the packets in the ERSPAN traffic, and saves it in the running configuration.
Step 12	**ip dscp** *dscp_value* **Example:** n1000v(config-monitor-erspan-src)# **ip dscp 24** n1000v(config-monitor-erspan-src)#	(Optional) Specifies the IP DSCP value, from 0-63. for the packets in the ERSPAN traffic, and saves it in the running configuration.
Step 13	**mtu** *mtu_value* **Example:** n1000v(config-monitor-erspan-src)# **mtu 1000** n1000v(config-monitor-erspan-src)#	(Optional) Specifies an MTU size for the ERSPAN traffic, and saves it in the running configuration.
Step 14	**erspan-id** *flow_id* **Example:** n1000v(config-erspan-src)# erspan-id 51	Adds an ERSPAN ID (1-1023) to the session configuration and saves it in the running configuration. The session ERSPAN ID is added to the ERSPAN header of the encapsulated frame and can be used at the termination box to differentiate between various ERSPAN streams of traffic.
Step 15	no shut **Example:** n1000v(config-erspan-src)# no shut	Enables the ERSPAN session and saves it in the running configuration. By default, the session is created in the shut state.
Step 16	**show monitor session** *session_id* **Example:** n1000v(config-erspan-src)# show monitor session 3	(Optional) Displays the ERSPAN session configuration as it exists in the running configuration.

	Command	Purpose

Step 17	`exit`
	
	Example:
	`n1000v(config-erspan-src)# exit`
	`n1000v(config)#`
		(Optional) Exits ERSPAN Source Configuration mode and returns you to CLI Configuration mode.

Step 18	**copy running-config startup-config**
	
	Example:
	`n1000v(config)# copy running-config`
	`startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Shutting Down a SPAN Session

Use this procedure to discontinue the copying of packets for a SPAN session. You can discontinue copying packets from one souce and destination; and then resume for another source and destination.

BEFORE YOU BEGIN

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

- You are logged in to the CLI in EXEC mode.
- You know which SPAN session that you want to shut down.
- You can shut down a SPAN session from either Global Configuration mode or Monitor Configuration mode.

SUMMARY STEPS

From Global Configuration mode:

1. **config t**
2. **monitor session** {*session-number* | *session-range* | **all**} **shut**
3. **show monitor**
4. **copy running-config startup-config**

From Monitor Configuration mode:

1. **config t**
2. **monitor session** {*session-number* | *session-range* | **all**} **[type erspan-source]**
3. **shut**
4. **show monitor**
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 2	**monitor session** {*session-number* | *session-range* | **all**} **shut**
	
	Example:
	`n1000v(config)# monitor session 3 shut`
	`n1000v(config)#`
	
	Example:
	`n1000v(config)# monitor session 3`
	`n1000v(config-monitor)# shut`
		Shuts down the specified SPAN monitor session(s)) from either Global Configuration mode or Monitor-Configuration mode.
		
		• session-number: Specifies a particular SPAN session number.
		
		• session range: Specifies a range of SPAN sessions (allowable = from 1 to 16).
		
		• all: Specifies all SPAN monitor sessions.

Step 3	**show monitor** **Example:** `n1000v(config-monitor)# show monitor`	(Optional) Displays the status of the SPAN sessions.

Step 4	**copy running-config startup-config**
	
	Example:
	`n1000v(config-monitor)# copy`
	`running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Resuming a SPAN Session

Use this procedure to resume the copying of packets for a SPAN session. You can discontinue copying packets from one souce and destination; and then resume for another source and destination.

BEFORE YOU BEGIN

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

• You are logged in to the CLI in EXEC mode.

• You know which SPAN session that you want to configure.

• You can resume the SPAN session from either Global Configuration mode or Monitor Configuration mode.

SUMMARY STEPS

From Global Configuration mode:

1. **config t**

2. **no monitor session** {*session-number* \| *session-range* \| **all**} **shut**

3. **show monitor**

4. **copy running-config startup-config**

From Monitor Configuration mode:

1. **config t**

2. **monitor session** {*session-number* | *session-range* | **all**} **[type erspan-source]**

3. **no shut**

4. **show monitor**

5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 2	**[no]** **monitor session** {*session-number session-range*	**all**} **shut**
		
		Example:
		`n1000v(config)# no monitor session 3 shut`
		`n1000v(config)#`
		
		Example:
		`n1000v(config)# monitor session 3`
		`n1000v(config-monitor)# no shut`
			Starts the specified SPAN monitor session(s) from either Global Configuration mode or Monitor-Configuration mode.
			
			• session-number: Specifies a particular SPAN session number.
			
			• session range: Specifies a range of SPAN sessions (allowable = from 1 to 16).
			
			• all: Specifies all SPAN monitor sessions.

Step 3	**show monitor** **Example:** `n1000v(config-monitor)# show monitor`	(Optional) Displays the status of the SPAN sessions.
Step 4	**copy running-config startup-config** **Example:** `n1000v(config-monitor)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Configuring the Allowable ERSPAN Flow IDs

Use this procedure to restrict the allowable range of flow IDs that can be assigned to ERSPAN sessions.

BEFORE YOU BEGIN

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

• You are logged in to the CLI in EXEC mode.

• You know the restricted range of ERSPAN flow IDs that you want to designate.

• The available ERSPAN flow IDs are 1-1024. You can restrict the range of available IDs using this procedure.

SUMMARY STEPS

1. **config t**
2. **[no] limit-resource erspan-flow-id minimum** *min_val* **maximum** *max_val*
3. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 2	**limit-resource erspan-flow-id minimum** *min_val* **maximum** *max_val*
	
	Example:
	`n1000v(config)# limit-resource`
	`erspan-flow-id minimum 20 maximum 40`
	`n1000v(config)#`
		Restricts the allowable range of ERSPAN flow IDs that can be assigned.

Step 3	**copy running-config startup-config**
	
	Example:
	`n1000v(config-monitor)# copy`
	`running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Verifying the SPAN Configuration

To verify the SPAN configuration, use the following commands:

Command	Purpose
show monitor session {**all** | *session-number* | **range** *session-range*} [**brief**]	Displays the SPAN session configuration.
show monitor	Displays Ethernet SPAN information.
module vem *module-number* **execute vemcmd show span**	Displays the configured SPAN sessions on a VEM module.
show port-profile name *port_profile_name*	Displays the Layer 3 capable port profile that is required for ERSPAN.

Example Configurations

This section includes the following example configurations:

-
-

Example Configuration for a SPAN Session

To configure a SPAN session, follow these steps:

Step 1   Configure destination ports in access or trunk mode, and enable SPAN monitoring.

```
n1000v# config t
  n1000v(config)# interface ethernet 2/5
    n1000v(config-if)# switchport
    n1000v(config-if)# switchport mode trunk
    n1000v(config-if)# no shut
    n1000v(config-if)# exit
  n1000v(config)#
```

Step 2   Configure a SPAN session.

```
n1000v(config)# no monitor session 3
n1000v(config)# monitor session 3
  n1000v(config-monitor)# source interface ethernet 2/1-3
  n1000v(config-monitor)# source interface port-channel 2
  n1000v(config-monitor)# source vlan 3, 6-8 tx
  n1000v(config-monitor)# filter vlan 3-5, 7
  n1000v(config-monitor)# destination interface ethernet 2/5
  n1000v(config-monitor)# no shut
  n1000v(config-monitor)# exit
n1000v(config)# show monitor session 3
n1000v(config)# copy running-config startup-config
```

Example Configuration for an ERSPAN Session

The following example shows how to create a bidirectional ERSPAN session for a source Ethernet interface and destination IP address. Packets arriving at the destination IP are identified by the ID 999 in their header.

```
n1000v(config)# monitor session 1 type erspan-source
n1000v(config-erspan-src)# source interface ethernet 3/3
n1000v(config-erspan-src)# destination ip 10.54.54.1
n1000v(config-erspan-src)# erspan-id 999
n1000v(config-erspan-src)# mtu 1000
n1000v(config-erspan-src)# no shut

n1000v(config)# show monitor session 1
   session 1
---------------
type            : erspan-source
state           : up
source intf     :
    rx          : Eth3/3
    tx          : Eth3/3
    both        : Eth3/3
source VLANs    :
    rx          :
    tx          :
    both        :
filter VLANs    : filter not specified
destination IP  : 10.54.54.1
ERSPAN ID       : 999
ERSPAN TTL      : 64
```

```
                  ERSPAN IP Prec.   : 0
                  ERSPAN DSCP       : 0
                  ERSPAN MTU        : 1000

                  n1000v# module vem 3 execute vemcmd show span

                  VEM SOURCE IP: 10.54.54.10

                  HW SSN ID           DST LTL/IP     ERSPAN ID
                          0           10.54.54.1           999
                          1                   48         local
```

Additional References

For additional information related to implementing SPAN, see the following sections:

Related Documents

Related Topic	Document Title
Port profile configuration	*Cisco Nexus 1000V Port Profile Configuration Guide, Release 4.0(4)SV1(3)*
Interface configuration	*Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(3)*
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(3)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for SPAN and ERSPAN

This section provides the SPAN and ERSPAN feature release history.

Feature Name	Releases	Feature Information
NAM support for ERSPAN data sources	4.0(4)SV1(3)	NAM support was introduced.

Feature Name	Releases	Feature Information
ERSPAN Type III header	4.0(4)SV1(3)	ERSPAN Type III header format was introduced.
SPAN and ERSPAN	4.0(4)SV1(1)	SPAN and ERSPAN were introduced.

Configuring SNMP

This chapter describes how to configure the SNMP including users, message encryption, notifications, authentication over TCP, and so forth.

Information About SNMP

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

SNMP Functional Overview

The SNMP framework consists of three parts:

- An SNMP manager—The system used to control and monitor the activities of network devices using SNMP.

- An SNMP agent—The software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems. Cisco Nexus 1000V supports the agent and MIB. To enable the SNMP agent, you must define the relationship between the manager and the agent.

- A managed information base (MIB)—The collection of managed objects on the SNMP agent.

SNMP is defined in RFCs 3411 to 3418.

Note SNMP sets are not supported.

Note SNMP Role Based Access Control (RBAC) is not supported.

SNMPv1, SNMPv2c, and SNMPv3. Both SNMPv1 and SNMPv2c use a community-based form of security are supported.

SNMP Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of a connection to a neighbor router, or other significant events.

SNMP notifications are generated as either traps or informs. A trap is an asynchronous, unacknowledged message sent from the agent to the SNMP managers listed in the host receiver table. Informs are asynchronous messages sent from the SNMP agent to the SNMP manager which the manager must acknowledge receipt of.

Traps are less reliable than informs because the SNMP manager does not send any acknowledgment when it receives a trap. Cisco Nexus 1000V cannot determine if the trap was received. An SNMP manager that receives an inform request acknowledges the message with an SNMP response protocol data unit (PDU). If Cisco Nexus 1000V never receives a response, it can send the inform request again.

You can configure Cisco Nexus 1000V to send notifications to multiple host receivers. See the "Configuring SNMP Notification Receivers" section on page 10-7 for more information about host receivers.

SNMPv3

SNMPv3 provides secure access to devices by a combination of authenticating and encrypting frames over the network. The security features provided in SNMPv3 are as follows:

- Message integrity—Ensures that a packet has not been tampered with in-transit.
- Authentication—Determines the message is from a valid source.
- Encryption—Scrambles the packet contents to prevent it from being seen by unauthorized sources.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy that is set up for a user and the role in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

This section includes the following topics:

- Security Models and Levels for SNMPv1, v2, v3, page 10-2
- User-Based Security Model, page 10-3
- CLI and SNMP User Synchronization, page 10-4
- Group-Based SNMP Access, page 10-4

Security Models and Levels for SNMPv1, v2, v3

The security level determines if an SNMP message needs to be protected from disclosure and if the message needs to be authenticated. The various security levels that exist within a security model are as follows:

- noAuthNoPriv—Security level that does not provide authentication or encryption.
- authNoPriv—Security level that provides authentication but does not provide encryption.
- authPriv—Security level that provides both authentication and encryption.

Three security models are available: SNMPv1, SNMPv2c, and SNMPv3. The security model combined with the security level determine the security mechanism applied when the SNMP message is processed.

Table 10-1 identifies what the combinations of security models and levels mean.

Table 10-1   SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	HMAC-MD5 or HMAC-SHA	No	Provides authentication based on the Hash-Based Message Authentication Code (HMAC) Message Digest 5 (MD5) algorithm or the HMAC Secure Hash Algorithm (SHA).
v3	authPriv	HMAC-MD5 or HMAC-SHA	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides Data Encryption Standard (DES) 56-bit encryption in addition to authentication based on the Cipher Block Chaining (CBC) DES (DES-56) standard.

User-Based Security Model

SNMPv3 User-Based Security Model (USM) refers to SNMP message-level security and offers the following services:

- Message integrity—Ensures that messages have not been altered or destroyed in an unauthorized manner and that data sequences have not been altered to an extent greater than can occur nonmaliciously.

- Message origin authentication—Ensures that the claimed identity of the user on whose behalf received data was originated is confirmed.

- Message confidentiality—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

SNMPv3 authorizes management operations only by configured users and encrypts SNMP messages.

Cisco Nexus 1000V uses two authentication protocols for SNMPv3:

- HMAC-MD5-96 authentication protocol

- HMAC-SHA-96 authentication protocol

Cisco Nexus 1000V uses Advanced Encryption Standard (AES) as one of the privacy protocols for SNMPv3 message encryption and conforms with RFC 3826.

The **priv** option offers a choice of DES or 128-bit AES encryption for SNMP security encryption. The **priv** option along with the **aes-128** token indicates that this privacy password is for generating a 128-bit AES key.The AES priv password can have a minimum of eight characters. If the passphrases are specified in clear text, you can specify a maximum of 64 case-sensitive alphanumeric characters. If you use the localized key, you can specify a maximum of 130 characters.

> **Note** For an SNMPv3 operation that uses the external AAA server, you must use AES for the privacy protocol in the user configuration on the external AAA server.

CLI and SNMP User Synchronization

SNMPv3 user management can be centralized at the Access Authentication and Accounting (AAA) server level. This centralized user management allows the SNMP agent in Cisco Nexus 1000V to leverage the user authentication service of the AAA server. Once user authentication is verified, the SNMP PDUs are processed further. Additionally, the AAA server is also used to store user group names. SNMP uses the group names to apply the access/role policy that is locally available in the switch.

Any configuration changes made to the user group, role, or password results in database synchronization for both SNMP and AAA.

Cisco Nexus 1000V synchronizes user configuration in the following ways:

- The authentication passphrase specified in the **snmp-server user** command becomes the password for the CLI user.
- The password specified in the **username** command becomes as the authentication and privacy passphrases for the SNMP user.
- If you delete a user using either SNMP or the CLI, the user is deleted for both SNMP and the CLI.
- User-role mapping changes are synchronized in SNMP and the CLI.
- Role changes (deletions or modifications) from the CLI are synchronized to SNMP.

> **Note** When you configure passphrase/password in localized key/encrypted format, Cisco Nexus 1000V does not synchronize the password.

Cisco NX-OS holds the synchronized user configuration for 60 minutes by default. See the "Modifying the AAA Synchronization Time" section on page 10-12 for information on how to modify this default value.

Group-Based SNMP Access

> **Note** Because *group* is a standard SNMP term used industry-wide, we refer to role(s) as group(s) in this SNMP section.

SNMP access rights are organized by groups. Each group in SNMP is similar to a role through the CLI. Each group is defined with read access or read-write access.

You can begin communicating with the agent once your username is created, your roles are set up by your administrator, and you are added to the roles.

High Availability

Stateless restarts for SNMP are supported. After a reboot or supervisor switchover, the running configuration is applied.

Guidelines and Limitations

SNMP has the following configuration guidelines and limitations:

- Read-only access to some SNMP MIBs is supported. See the Cisco NX-OS MIB support list at the following URL for more information:

  http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

- SNMP Role Based Access Control (RBAC) is not supported.

- SNMP sets are not supported.

Default Settings

Table 10-2 lists the default settings for SNMP parameters.

Table 10-2   Default SNMP Parameters

Parameters	Default
license notifications	enabled

Configuring SNMP

This section includes the following topics:

Note Be aware that the Cisco NX-OS commands for this feature may differ from those used in Cisco IOS.

Configuring SNMP Users

Use this procedure to configure a user for SNMP.

BEFORE YOU BEGIN

• You are logged in to the CLI in EXEC mode.

SUMMARY STEPS

1. **config t**
2. **snmp-server user** *name* [**auth** {**md5** | **sha**} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID** *id*] [**localizedkey**]]
3. **show snmp user**
4. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**config t** **Example:** switch# config t Enter configuration commands, one per line. End with CNTL/Z. switch(config)#	Places you in global configuration mode.
Step 2	**snmp-server user** *name* [**auth** {**md5** | **sha**} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID** *id*] [**localizedkey**]] **Example:** switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh	Configures an SNMP user with authentication and privacy parameters. The passphrase can be any case-sensitive alphanumeric string up to 64 characters. If you use the **localizekey** keyword, the passphrase an be any case-sensitive alphanumeric string up to 130 characters. The engineID format is a 12-digit colon-separated decimal number.
Step 3	**show snmp user** **Example:** switch(config-callhome)# show snmp user	(Optional) Displays information about one or more SNMP users.
Step 4	**copy running-config startup-config** **Example:** switch(config)# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

This example shows how to configure the SNMP contact and location information:

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh
```

Enforcing SNMP Message Encryption

You can configure SNMP to require authentication or encryption for incoming requests. By default, the SNMP agent accepts SNMPv3 messages without authentication and encryption. When you enforce privacy, Cisco Nexus 1000V responds with an authorizationError for any SNMPv3 PDU request using securityLevel parameter of either noAuthNoPriv or authNoPriv.

Use the following command in global configuration mode to enforce SNMP message encryption for a user:

Command	Purpose
snmp-server user *name* **enforcePriv** **Example:** switch(config)# snmp-server user Admin enforcePriv	Enforces SNMP message encryption for this user.

Use the following command in global configuration mode to enforce SNMP message encryption for all users:

Command	Purpose
snmp-server globalEnforcePriv **Example:** switch(config)# snmp-server globalEnforcePriv	Enforces SNMP message encryption for all users.

Creating SNMP Communities

You can create SNMP communities for SNMPv1 or SNMPv2c.

Use the following command in global configuration mode to create an SNMP community string:

Command	Purpose
snmp-server community *name* {**ro**	**rw**} **Example:** switch(config)# snmp-server community public ro	Creates an SNMP community string.

Configuring SNMP Notification Receivers

You can configure Cisco Nexus 1000V to generate SNMP notifications to multiple host receivers.

Use the following command in global configuration mode to configure a host receiver for SNMPv1 traps:

Command	Purpose
snmp-server host *ip-address* **traps version 1** *community* [**udp_port** *number*] **Example:** switch(config)# snmp-server host 192.0.2.1 traps version 1 public	Configures a host receiver for SNMPv1 traps. The community can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.

Use the following command in global configuration mode to configure a host receiver for SNMPv2c traps or informs:

Command	Purpose
snmp-server host *ip-address* {**traps** | **informs**} **version 2c** *community* [**udp_port** *number*] **Example:** switch(config)# snmp-server host 192.0.2.1 informs version 2c public	Configures a host receiver for SNMPv2c traps or informs. The community can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.

Use the following command in global configuration mode to configure a host receiver for SNMPv3 traps or informs:

Command	Purpose
snmp-server host *ip-address* {**traps** | **informs**} **version 3** {**auth** | **noauth** | **priv**} *username* [**udp_port** *number*] **Example:** switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS	Configures a host receiver for SNMPv2c traps or informs. The username can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.

Note The SNMP manager must know the user credentials (authKey/PrivKey) based on the SNMP engine ID of the Cisco Nexus 1000V device to authenticate and decrypt the SNMPv3 messages.

Configuring the Notification Target User

You must configure a notification target user on the device to send SNMPv3 inform notifications to a notification host receiver.

Cisco Nexus 1000V uses the credentials of the notification target user to encrypt the SNMPv3 inform notification messages to the configured notification host receiver.

Note For authenticating and decrypting the received INFORM PDU, the notification host receiver should have the same user credentials as configured in Cisco Nexus 1000V to authenticate and decrypt the inform s.

Use the following command in global configuration mode to configure the notification target user:

Command	Purpose
`snmp-server user` *name* [**auth** {md5 | sha} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID** *id*] **Example:** `switch(config)# snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID 00:00:00:63:00:01:00:10:20:15:10:03`	Configures the notification target user with the specified engine ID for notification host receiver. The engineID format is a 12-digit colon-separated decimal number.

Enabling SNMP Notifications

You can enable or disable notifications. If you do not specify a notification name, Cisco Nexus 1000V enables all notifications.

Table 10-3 lists the commands that enable the notifications for Cisco Nexus 1000V MIBs.

Note   The **snmp-server enable traps** command enables both traps and informs, depending on the configured notification host receivers.

Table 10-3   Enabling SNMP Notifications

MIB	Related Commands
All notifications	**snmp-server enable traps**
CISCO-AAA-SERVER-MIB	**snmp-server enable traps aaa**
ENITY-MIB	**snmp-server enable traps entity**
CISCO-ENTITY-FRU-CONTROL-MIB	**snmp-server enable traps entity fru**
CISCO-LICENSE-MGR-MIB	**snmp-server enable traps license**
IF-MIB	**snmp-server enable traps link**
CISCO-PSM-MIB	**snmp-server enable traps port-security**
SNMPv2-MIB	**snmp-server enable traps snmp** **snmp-server enable traps snmp authentication**

The license notifications are enabled by default. All other notifications are disabled by default.

Use the following commands in global configuration mode to enable the specified notification:

Command	Purpose
snmp-server enable traps **Example:** switch(config)# snmp-server enable traps	Enables all SNMP notifications.
snmp-server enable traps aaa [**server-state-change**] **Example:** switch(config)# snmp-server enable traps aaa	Enables the AAA SNMP notifications.
snmp-server enable traps entity [**fru**] **Example:** switch(config)# snmp-server enable traps entity	Enables the ENTITY-MIB SNMP notifications.
snmp-server enable traps license **Example:** switch(config)# snmp-server enable traps license	Enables the license SNMP notification.
snmp-server enable traps link **Example:** switch(config)# snmp-server enable traps link	Enables the link SNMP notifications.
snmp-server enable traps port-security **Example:** switch(config)# snmp-server enable traps port-security	Enables the port security SNMP notifications.
snmp-server enable traps snmp [**authentication**] **Example:** switch(config)# snmp-server enable traps snmp	Enables the SNMP agent notifications.

Disabling LinkUp/LinkDown Notifications on an Interface

You can disable linkUp and linkDown notifications on an individual interface. You can use this limit notifications on flapping interface (an interface that transitions between up and down repeatedly).

Use the following command in interface configuration mode to disable linkUp/linkDown notifications for the interface:

Command	Purpose
no snmp trap link-status **Example:** switch(config-if)# no snmp trap link-status	Disables SNMP link-state traps for the interface. This command is enabled by default.

Enabling a One-time Authentication for SNMP over TCP

You can enable a one-time authentication for SNMP over a TCP session.

Use the following command in global configuration mode to enable one-time authentication for SNMP over TCP:

Command	Purpose
`snmp-server tcp-session` [`auth`] **Example:** `switch(config)# snmp-server tcp-session`	Enables a one-time authentication for SNMP over a TCP session. The default is disabled.

Assigning the SNMP Switch Contact and Location Information

You can assign the switch contact information, which is limited to 32 characters (without spaces) and the switch location.

BEFORE YOU BEGIN

- You are logged in to the CLI in EXEC mode.

SUMMARY STEPS

1. **config t**
2. **snmp-server contact** *name*
3. **snmp-server location** *name*
4. **show snmp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`switch# config t`
	`Enter configuration commands, one per`
	`line.  End with CNTL/Z.`
	`switch(config)#`
		Places you in global configuration mode.

Step 2	`snmp-server contact` *name*
	
	Example:
	`switch(config)# snmp-server contact`
	`Admin`
		Configures sysContact, which is the SNMP contact name.

Step 3	`snmp-server location` *name*
	
	Example:
	`switch(config)# snmp-server location`
	`Lab-7`
		Configures sysLocation, which is the SNMP location.

	Command	Purpose
Step 4	`show snmp` **Example:** switch(config)# show snmp	(Optional) Displays information about one or more destination profiles.
Step 5	`copy running-config startup-config` **Example:** switch(config)# copy running-config startup-config	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

This example shows how to configure the SNMP contact and location information:

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp contact Admin
switch(config)# snmp location Lab-7
```

Disabling SNMP

You can disable the SNMP protocol on a device.

Use the following command in global configuration mode to disable the SNMP protocol

Command	Purpose
`no snmp-server protocol enable` **Example:** switch(config)# no snmp-server protocol enable	Disables the SNMP protocol. This command is enabled by default.

Modifying the AAA Synchronization Time

You can modify how long Cisco NX-OS holds the synchronized user configuration.

Use the following command in global configuration mode to modify the AAA synchronization time:

Command	Purpose
`snmp-server aaa-user cache-timeout` *seconds* **Example:** switch(config)# snmp-server aaa-user cache-timeout 1200.	Configures how long the AAA synchronized user configuration stays in the local cache. The range is from 1 to 86400 seconds. The default is 3600.

Verifying the SNMP Configuration

To display the SNMP configuration, use the following commands:

Command	Purpose
show running-config snmp [**all**]	Displays the SNMP running configuration.
show snmp	Displays the SNMP status.
show snmp community	Displays the SNMP community strings.
show snmp context	Displays the SNMP context mapping.
show snmp engineID	Displays the SNMP engineID.
show snmp group	Displays SNMP roles.
show snmp session	Displays SNMP sessions.
show snmp trap	Displays the SNMP notifications enabled or disabled.
show snmp user	Displays SNMPv3 users.

SNMP Example Configuration

This example configures sending the Cisco linkUp/Down notifications to one notification host receiver using the Blue VRF and defines two SNMP users, Admin and NMS:

```
config t
 snmp-server contact Admin@company.com
 snmp-server user Admin auth sha abcd1234 priv abcdefgh
 snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID
00:00:00:63:00:01:00:22:32:15:10:03
 snmp-server host 192.0.2.1 informs version 3 auth NMS
 snmp-server host 192.0.2.1 use-vrf Blue
 snmp-server enable traps link cisco
```

Additional References

For additional information related to implementing SNMP, see the following sections:

- Related Documents, page 10-13
- Standards, page 10-14
- MIBs, page 10-14

Related Documents

Related Topic	Document Title
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(3)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
• SNMP-COMMUNITY-MIB • SNMP-FRAMEWORK-MIB • SNMP-NOTIFICATION-MIB • SNMP-TARGET-MIB • SNMPv2-MIB	To locate and download MIBs, go to the following URL: http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

Feature History for SNMP

This section provides the SNMP feature release history.

Feature Name	Releases	Feature Information
SNMP	4.0(4)SV1(1)	This feature was introduced.

Configuring NetFlow

Use this chapter to configure NetFlow to characterize IP traffic based on its source, destination, timing, and application information, to assess network availability and performance.
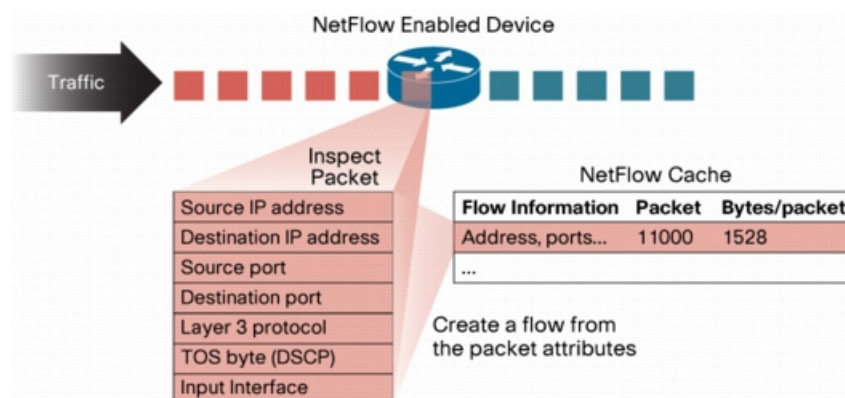
Information About NetFlow

NetFlow lets you evaluate IP traffic and understand how and where it flows. NetFlow gathers data that can be used in accounting, network monitoring, and network planning.

What is a Flow

A flow is a one-directional stream of packets that arrives on a source interface (or subinterface), matching a set of criteria. All packets with the same source/destination IP address, source/destination ports, protocol interface and class of service are grouped into a flow and then packets and bytes are tallied. This condenses a large amount of network information into a database called the NetFlow cache.

Figure 11-1   Creating a Flow in the NetFlow Cache

You create a flow by defining the criteria it gathers. Flows are stored in the NetFlow cache.

Flow information tells you the following:

- Source address tells you who is originating the traffic.

- Destination address tells who is receiving the traffic.
- Ports characterize the application using the traffic.
- Class of service examines the priority of the traffic.
- The device interface tells how traffic is being used by the network device.
- Tallied packets and bytes show the amount of traffic.

Flow Record Definition

A flow record defines the information that NetFlow gathers, such as packets in the flow and the types of counters gathered per flow. You can define new flow records or use the predefined Cisco Nexus 1000V flow record.

To create a record, see the "Defining a Flow Record" procedure on page 11-10.

The following table describes the criteria defined in a flow record.

Flow record criteria	Description

Match	Defines what information is matched for collection in the flow record.
	
	 • **ip**: Data collected in the flow record matches one of the following IP options:
	    – protocol
	    – tos (type of service)
	
	 • **ipv4**: Data collected in the flow record matches one of the following ipv4 address options:
	    – source address
	    – destination adress
	
	 • **transport**: Data collected in the flow record matches one of the following transport options:
	    – destination port
	    – source port

Collect	Defines how the flow record collects information.
	
	 • **counter**: Collects Flow Record information in one of the following formats:
	    – bytes: collected in 32-bit counters unless the long 64-bit counter is specified.
	    – packets: collected in 32-bit counters unless the long 64-bit counter is specified.
	
	 • **timestamp sys-uptime**: Collects the system up time for the first or last packet in the flow.
	
	 • **transport tcp flags**: Collects the TCP transport layer flags for the packets in the flow.

Predefined Flow Records

Cisco Nexus 1000V includes the following pre-defined flow records.

Example 11-1   Cisco Nexus 1000V Predefined Flow Record: Netflow-Original

```
n1000v# show flow record netflow-original
Flow record netflow-original:
    Description: Traditional IPv4 input NetFlow with origin ASs
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 source address
        match ipv4 destination address
        match ip protocol
        match ip tos
        match transport source-port
        match transport destination-port
        match interface input
        match interface output
        match flow direction
        collect routing source as
        collect routing destination as
        collect routing next-hop address ipv4
        collect transport tcp flags
        collect counter bytes
        collect counter packets
        collect timestamp sys-uptime first
        collect timestamp sys-uptime last
n1000v#
```

✎

Note   Although the following lines appear in the output of the **show flow record** command, the commands they are based on are not currently supported in Cisco Nexus 1000V. The use of these commands has no affect on the configuration.

```
collect routing source as
collect routing destination as
collect routing next-hop address ipv4
```

Example 11-2   Cisco Nexus 1000V Predefined Flow Record: Netflow IPv4 Original-Input

```
n1000v# show flow record netflow ipv4 original-input
Flow record ipv4 original-input:
    Description: Traditional IPv4 input NetFlow
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 source address
        match ipv4 destination address
        match ip protocol
        match ip tos
        match transport source-port
        match transport destination-port
        match interface input
        match interface output
        match flow direction
        collect routing source as
        collect routing destination as
        collect routing next-hop address ipv4
```

```
        collect transport tcp flags
        collect counter bytes
        collect counter packets
        collect timestamp sys-uptime first
        collect timestamp sys-uptime last
n1000v#
```

Example 11-3   Cisco Nexus 1000V Predefined Flow Record: Netflow IPv4 Original-Output

```
switch# show flow record netflow ipv4 original-output
Flow record ipv4 original-output:
    Description: Traditional IPv4 output NetFlow
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 source address
        match ipv4 destination address
        match ip protocol
        match ip tos
        match transport source-port
        match transport destination-port
        match interface input
        match interface output
        match flow direction
        collect routing source as
        collect routing destination as
        collect routing next-hop address ipv4
        collect transport tcp flags
        collect counter bytes
        collect counter packets
        collect timestamp sys-uptime first
        collect timestamp sys-uptime last
switch#
```

Example 11-4   Cisco Nexus 1000V Predefined Flow Record: Netflow IPv4 Protocol-Port

```
switch# show flow record netflow ipv4 protocol-port
Flow record ipv4 protocol-port:
    Description: Protocol and Ports aggregation scheme
    No. of users: 0
    Template ID: 0
    Fields:
        match ip protocol
        match transport source-port
        match transport destination-port
        match interface input
        match interface output
        match flow direction
        collect counter bytes
        collect counter packets
        collect timestamp sys-uptime first
        collect timestamp sys-uptime last
switch#
```

Accessing NetFlow Data

There are two primary methods used to access NetFlow data:

- Command Line Interface (CLI), page 11-5
- NetFlow Collector, page 11-6

Command Line Interface (CLI)

To view what is happening in your network now, use the CLI. To see a list of available show commands, see the "Verifying the NetFlow Configuration" section on page 11-18.

The CLI uses the following tools to capture and export flow records to the Netflow Collector:

- Flow Monitor, page 11-5
- Flow Exporter, page 11-5

Flow Monitor

A flow monitor creates an association between the following NetFlow components:

- a flow record—consisting of matching and collection criteria
- a flow exporter—consisting of the export criteria

This flow monitor association enables a set, consisting of a record and an exporter, to be defined once and re-used many times. Multiple flow monitors can be created for different needs. A flow monitor is applied to a specific interface in a specific direction.

See the "Defining a Flow Monitor" procedure on page 11-15, and "Assigning a Flow Monitor to an Interface" procedure on page 11-17.

Flow Exporter

Use the flow exporter to define where and when the flow records are sent from the cache to the reporting server, called the NetFlow Collector.

An exporter definition includes the following.

- Destination IP address
- Source interface
- UDP port number (where the collector is listening)
- Export format

Note   NetFlow export packets use the IP address assigned to the source interface. If the source interface does not have an IP address assigned to it, the exporter will be inactive.

See the "Defining a Flow Exporter" procedure on page 11-13.

Export Formats

Cisco Nexus 1000V supports the NetFlow Version 9 export format.

Note   Cisco Nexus 1000V supports UDP as the transport protocol for exporting data to up to two exporters per monitor.

NetFlow Collector

You can export NetFlow from the Cisco Nexus 1000V NetFlow cache to a reporting server called the NetFlow Collector. The NetFlow Collector assembles the exported flows and combines them to produce reports used for traffic and security analysis. NetFlow export, unlike SNMP polling, pushes information periodically to the NetFlow reporting collector. The NetFlow cache is constantly filling with flows. Cisco Nexus 1000V searches the cache for flows that have terminated or expired and exports them to the NetFlow collector server. Flows are terminated when the network communication has ended, that is, when a packet contains the TCP FIN flag.

The following steps implement NetFlow data reporting:

- NetFlow records are configured to define the information that NetFlow gathers.
- Netflow monitor is configured to capture flow records to the NetFlow cache.
- NetFlow export is configured to send flows to the collector.
- Cisco Nexus 1000V searches the NetFlow cache for flows that have terminated and exports them to the NetFlow collector server.
- Flows are bundled together based on space availability in the UDP export packet or based on export timer.
- The NetFlow collector software creates real-time or historical reports from the data.

Exporting Flows to the NetFlow Collector Server

Timers determine when a flow is exported to the NetFlow Collector Server.

A flow is ready for export when one of the following occurs:

- The flow is inactive for a certain time during which no new packets are received for the flow.
- The flow has lived longer than the active timer, for example, a long FTP download.
- A TCP flag indicates the flow is terminated. That is, a FIN or RST flag is present.
- The flow cache is full and some flows must be aged out to make room for new flows.

Figure 11-2   Exporting Flows to the NetFlow Collector Server
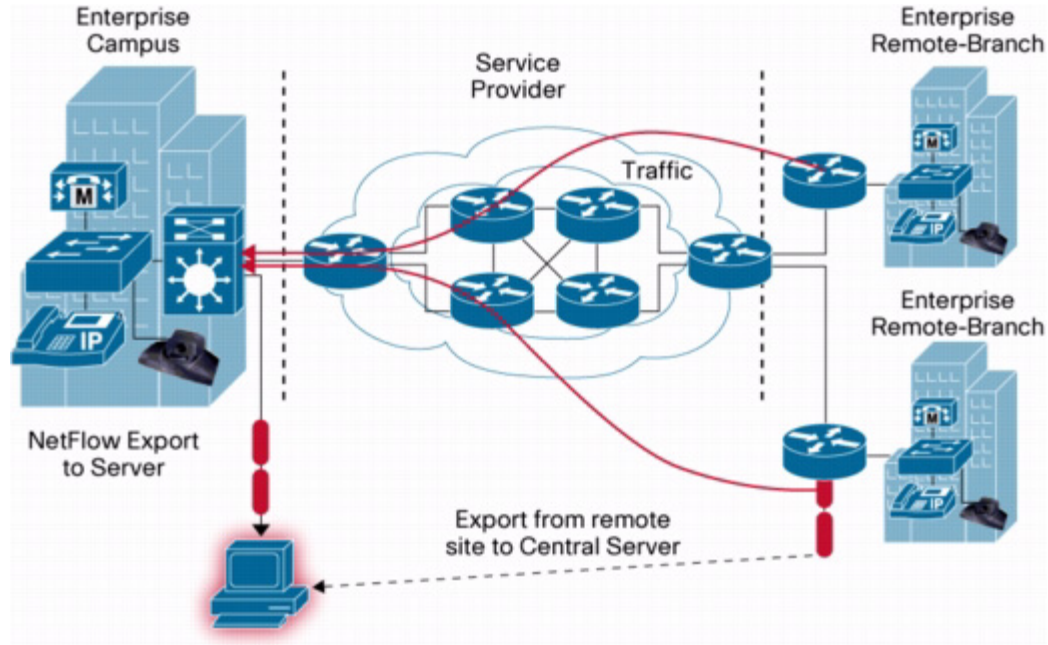
What NetFlow Data Looks Like

The following figure shows an example of NetFlow data.

Figure 11-3   NetFlow Cache Example

1. Flow cache—The first unique packet creates a flow

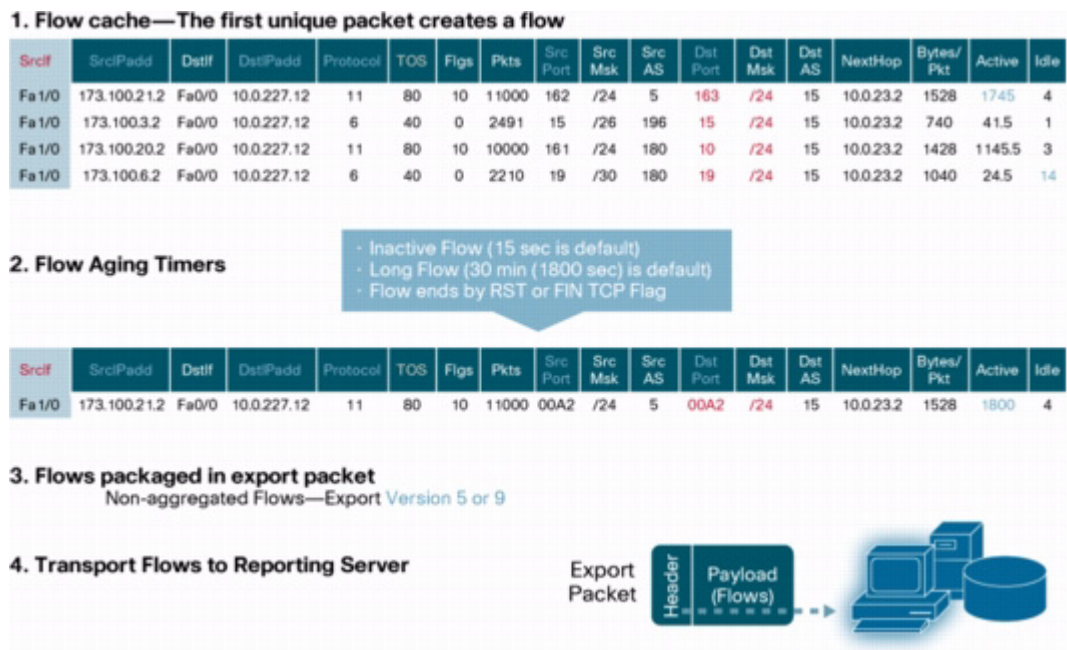| SrcIf	SrcIPadd	DstIf	DstIPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	162	/24	5	163	/24	15	10.0.23.2	1528	1745	4
Fa1/0	173.100.3.2	Fa0/0	10.0.227.12	6	40	0	2491	15	/26	196	15	/24	15	10.0.23.2	740	41.5	1
Fa1/0	173.100.20.2	Fa0/0	10.0.227.12	11	80	10	10000	161	/24	180	10	/24	15	10.0.23.2	1428	1145.5	3
Fa1/0	173.100.6.2	Fa0/0	10.0.227.12	6	40	0	2210	19	/30	180	19	/24	15	10.0.23.2	1040	24.5	14

2. Flow Aging Timers

- Inactive Flow (15 sec is default)
- Long Flow (30 min (1800 sec) is default)
- Flow ends by RST or FIN TCP Flag

SrcIf	SrcIPadd	DstIf	DstIPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1800	4

3. Flows packaged in export packet
Non-aggregated Flows—Export Version 5 or 9

4. Transport Flows to Reporting Server

Export Packet — Header — Payload (Flows)

Network Analysis Module

You can also use the Cisco Network Analysis Module (NAM) to monitor NetFlow data sources. NAM enables traffic analysis views and reports such as hosts, applications, conversations, VLAN, and QoS.

To use NAM for monitoring the Cisco Nexus 1000V NetFlow data sources see the *Cisco Nexus 1010 Network Analysis Module Installation and Configuration Note, 4.2*.

High Availability

Cisco Nexus 1000V supports stateful restarts for NetFlow. After a reboot or supervisor switchover, Cisco Nexus 1000V applies the running configuration.

Prerequisites for NetFlow

- You must be aware of resource requirements since NetFlow consumes additional memory and CPU resources.

- Memory and CPU resources are provided by the VEM hosting the flow monitor interface. Resources are limited by the number of CPU cores present on the VEM.

Configuration Guidelines and Limitations

NetFlow has the following configuration guidelines and limitations:

- If a source interface is not configured, the NetFlow exporter will remain disabled.

- In Cisco Nexus 1000V, Mgmt0 interface is configured by default as the source interface for an exporter. You can change the source interface if needed.

- Cisco Nexus 1000V includes the following predefined flow records that can be used instead of configuring a new one. For more information, see the "Flow Record Definition" section on page 11-2:

  - netflow-original
    Cisco Nexus 1000V predefined traditional IPv4 input NetFlow with origin ASs

    ✎
    Note   The routing-related fields in this predefined flow record are ignored.

  - netflow ipv4 original-input
    Cisco Nexus 1000V predefined traditional IPv4 input NetFlow

  - netflow ipv4 original-output
    Cisco Nexus 1000V predefined traditional IPv4 output NetFlow

  - netflow ipv4 protocol-port
    Cisco Nexus 1000V predefined `protocol and ports aggregation scheme`

- Up to 256 NetFlow interfaces are allowed per DVS.

- Up to 32 NetFlow interfaces are allowed per host

- A maximum of one flow monitor per interface per direction is allowed.

- Up to 8 flow monitors are allowed per VEM.

- Up to 2 flow exporters are permitted per monitor.

- Up to 32 NetFlow Policies are allowed per DVS.

- Up to 8 NetFlow Policies are allowed per host.

- NetFlow is not supported on port channels.

Default Settings

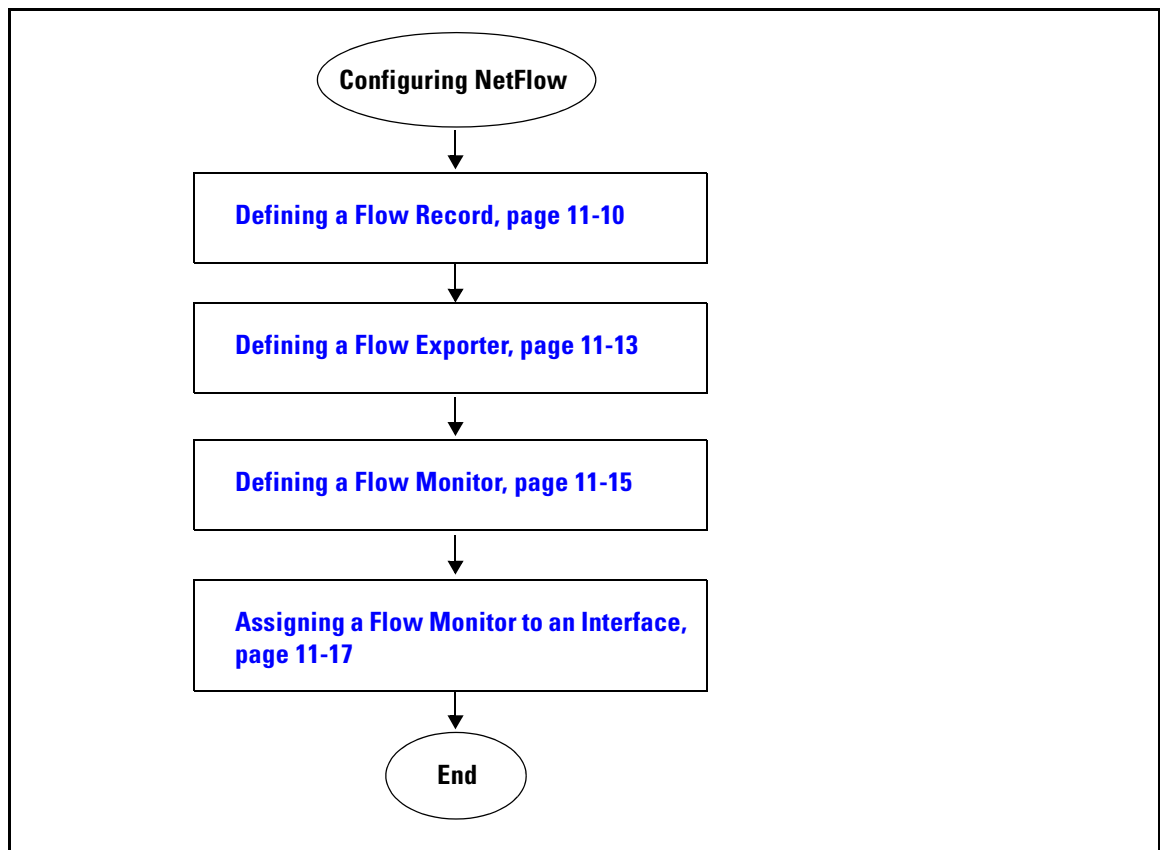Table 11-1 lists the default settings for NetFlow parameters.

Table 11-1   Default NetFlow Parameters

Parameters	Default
NetFlow version	9
source interface	mgmt0
match	direction and interface (incoming/outgoing)
flow monitor active timeout	1800
flow monitor inactive timeout	15
flow monitor cache size	4096
flow exporter UDP port **transport udp** command	9995
DSCP	default/best-effort (0)
VRF	default

Configuring NetFlow

The following flow chart is designed to guide you through the netflow configuration process. After completing each procedure, return to the flow chart to make sure you complete all required procedures in the correct sequence.

Flow Chart: Configuring NetFlow

```
                    ┌─────────────────────────┐
                    │   Configuring NetFlow    │
                    └─────────────────────────┘
                                 │
                                 ▼
            ┌──────────────────────────────────────────┐
            │   Defining a Flow Record, page 11-10       │
            └──────────────────────────────────────────┘
                                 │
                                 ▼
            ┌──────────────────────────────────────────┐
            │   Defining a Flow Exporter, page 11-13     │
            └──────────────────────────────────────────┘
                                 │
                                 ▼
            ┌──────────────────────────────────────────┐
            │   Defining a Flow Monitor, page 11-15      │
            └──────────────────────────────────────────┘
                                 │
                                 ▼
            ┌──────────────────────────────────────────┐
            │   Assigning a Flow Monitor to an Interface,│
            │   page 11-17                               │
            └──────────────────────────────────────────┘
                                 │
                                 ▼
                    ┌─────────────────────────┐
                    │           End           │
                    └─────────────────────────┘
```

Defining a Flow Record

Use this procedure to create a flow record.

Note Optionally, you can use the Cisco Nexus 1000V pre-defined record shown in the "Flow Record Definition" section on page 11-2. See the "Defining a Flow Monitor" section on page 11-15 to apply a pre-defined record to a flow monitor.

BEFORE YOU BEGIN

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

- You know which of the options you want this flow record to match.
- You know which options you want this flow record to collect.

    For more information, see the "Flow Record Definition" section on page 11-2.

Note Although the following lines appear in the output of the **show flow record** command, the commands they are based on are not currently supported in Cisco Nexus 1000V. The use of these commands has no affect on the configuration.

collect routing source as
collect routing destination as
collect routing next-hop address ipv4

SUMMARY STEPS

1. **config t**

2. **flow record** *name*

3. **description** *string*

4. **match {ip {protocol | tos} | ipv4 {destination address | source address} | transport {destination-port | source-port}}**

5. **collect {counter {bytes [long] | packets [long]} | timestamp sys-uptime | transport tcp flags}**

6. **show flow record** [*name*]

7. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	*config t*
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 2	**flow record** *name*
	
	Example:
	`n1000v(config)# flow record RecordTest`
	`n1000v(config-flow-record)#`
		Creates a Flow Record by name, and places you in the CLI Flow Record Configuration mode for that specific record.

Step 3	**description** *string* **Example:** `n1000v(config-flow-record)# description Ipv4Flow`	(Optional) Adds a description of up to 63 characters to this Flow Record and saves it in the running configuration.

Step 4	**match {ip{protocol| tos}|ipv4{destination address|source address}|transport {destination-port|source-port}}**
	
	Example:
	`n1000v(config-flow-record)# match ipv4 destination address`
		Defines the Flow Record to match one of the following and saves it in the running configuration.
		
		• **ip**: Matches one of the following IP options:
		  – protocol
		  – tos (type of service)
		
		• **ipv4**: Matches one of the following ipv4 address options:
		  – source address
		  – destination adress
		
		• **transport**: Matches one of the following transport options:
		  – destination port
		  – source port

	Command	Purpose

Step 5	collect {counter {bytes [long] | packets [long]} | timestamp sys-uptime | transport tcp flags}
	
	Example:
	`n1000v(config-flow-record)# collect counter packets`
		Specifies a collection option to define the information to collect in the Flow Record and saves it in the running configuration.
		
		• **counter**: Collects Flow Record information in one of the following formats:
		
		  – bytes: collected in 32-bit counters unless the long 64-bit counter is specified.
		
		  – packets: collected in 32-bit counters unless the long 64-bit counter is specified.
		
		• **timestamp sys-uptime**: Collects the system up time for the first or last packet in the flow.
		
		• **transport tcp flags**: Collects the TCP transport layer flags for the packets in the flow.

Step 6	show flow record [*name*] **Example:** `n1000v(config-flow-exporter)# show flow record RecordTest`	(Optional) Displays information about Flow Records.
Step 7	**copy running-config startup-config** **Example:** `n1000v(config-flow-exporter)# copy running-config startup-config`	(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

The following is an example for creating a flow record:

```
n1000v# config t
n1000v(config)# flow record RecordTest
n1000v(config-flow-record)# description Ipv4flow
n1000v(config-flow-record)# match ipv4 destination address
n1000v(config-flow-record)# collect counter packets
n1000v(config-flow-record)# show flow record RecordTest
Flow record RecordTest:
    Description: Ipv4flow
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 destination address
        match interface input
        match interface output
        match flow direction
        collect counter packets
n1000v(config-flow-record)#
```

Defining a Flow Exporter

Use this procedure to create a Flow Exporter defining where and how Flow Records are exported to the NetFlow Collector Server.

BEFORE YOU BEGIN

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

- A maximum of two flow exporters per monitor are permitted.
- You know destination IP address of the NetFlow Collector Server.
- You know the source interface that Flow Records are sent from.
- You know the transport UDP that the Collector is listening on.
- Export format version 9 is the version supported.

SUMMARY STEPS

1. **config t**
2. **flow exporter** *name*
3. **description** *string*
4. **destination** {*ipv4-address* | *ipv6-address*}
5. **dscp** *value*
6. **source mgmt** *interface_number*
7. **transport udp** *port-number*
8. **version 9**
9. **option {exporter-stats | interface-table} timeout** *seconds*
10. **template data timeout** *seconds*
11. **show flow exporter** [*name*]
12. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in CLI Global Configuration mode.

Step 2	**flow exporter** *name*
	
	Example:
	`n1000v(config)# flow exporter ExportTest`
	`n1000v(config-flow-exporter)#`
		Creates a Flow Exporter, saves it in the running configuration, and then places you in CLI Flow Exporter Configuration mode.

Step 3	**description** *string*
	
	Example:
	`n1000v(config-flow-exporter)#`
	`description ExportV9`
		Adds a description of up to 63 characters to this Flow Exporter and saves it in the running configuration.

	Command	Purpose

Step 4	**destination** {*ipv4-address* | *ipv6-address*}
	
	Example:
	`n1000v(config-flow-exporter)#`
	`destination 192.0.2.1`
		Specifies the IP address of the destination interface for this Flow Exporter and saves it in the running configuration.

Step 5	**dscp** *value* **Example:** `n1000v(config-flow-exporter)# dscp 0`	Specifies the differentiated services codepoint value for this Flow Exporter, between 0 and 63, and saves it in the running configuration.

Step 6	**source mgmt** *interface_number*
	
	Example:
	`n1000v(config-flow-exporter)# source`
	`mgmt 0`
		Specifies the interface and its number, from which the Flow Records are sent to the NetFlow Collector Server, and saves it in the running configuration.

Step 7	**transport udp** *port-number*
	
	Example:
	`n1000v(config-flow-exporter)# transport`
	`udp 200`
		Specifies the destination UDP port, between 0 and 65535, used to reach the NetFlow collecton, and saves it in the running configuration.

Step 8	**version** {**9**}
	
	Example:
	`n1000v(config-flow-exporter)# version 9`
	`n1000v(config-flow-exporter-version-9)#`
		Specifies NetFlow export version 9, saves it in the running configuration, and places you into the export version 9 configuration mode.

Step 9	**option** {**exporter-stats** | **interface-table** | **sampler-table**} **timeout** *value*
	
	Example:
	`n1000v(config-flow-exporter-version-9)#`
	`option exporter-stats timeout 1200`
		Specifies one of the following version 9 exporter resend timers and its value, between 1 and 86400 seconds, and saves it in the running configuration.
		
		• **exporter-stats**
		
		• **interface-table**
		
		• **sampler-tabl**e

Step 10	**template data timeout** *seconds*
	
	Example:
	`n1000v(config-flow-exporter-version-9)#`
	`template data timeout 1200`
		Sets the template data resend timer and its value, between 1 and 86400 seconds, and saves it in the running configuration.

Step 11	**show flow exporter** [*name*]
	
	Example:
	`n1000v(config-flow-exporter)# show flow`
	`exporter`
		(Optional) Displays information about the Flow Exporter.

Step 12	**copy running-config startup-config**
	
	Example:
	`n1000v(config-flow-exporter)# copy`
	`running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

The following is an example of creating a flow exporter:

```
n1000v(config)# flow exporter ExportTest
n1000v(config-flow-exporter)# description ExportHamilton
n1000v(config-flow-exporter)# destination 192.0.2.1
n1000v(config-flow-exporter)# dscp 2
n1000v(config-flow-exporter)# source mgmt 0
n1000v(config-flow-exporter)# transport udp 200
n1000v(config-flow-exporter)# version 9
n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200
n1000v(config-flow-exporter-version-9)# template data timeout 1200
```

```
n1000v(config-flow-exporter-version-9)# show flow exporter ExportTest
Flow exporter ExportTest:
    Description: ExportHamilton
    Destination: 192.0.2.1
    VRF: default (1)
    Destination UDP Port 200
    Source Interface Mgmt0
    DSCP 2
    Export Version 9
        Exporter-stats timeout 1200 seconds
        Data template timeout 1200 seconds
    Exporter Statistics
        Number of Flow Records Exported 0
        Number of Templates Exported 0
        Number of Export Packets Sent 0
        Number of Export Bytes Sent 0
        Number of Destination Unreachable Events 0
        Number of No Buffer Events 0
        Number of Packets Dropped (No Route to Host) 0
        Number of Packets Dropped (other) 0
        Number of Packets Dropped (LC to RP Error) 0
        Number of Packets Dropped (Output Drops) 1
        Time statistics were last cleared: Never
n1000v(config-flow-exporter-version-9)#
```

Defining a Flow Monitor

Use this procedure to create a Flow Monitor and associate a Flow Record and a Flow Exporter to it.

BEFORE YOU BEGIN

- A maximum of one flow monitor per interface per direction is permitted.
- You know the name of an existing Flow Exporter to associate with this flow monitor.
- You know the name of an existing Flow Record to associate with this flow monitor. You can use either a flow record you previously created, or one of the following Cisco Nexus 1000V predefined flow records:
  - netflow-original
  - netflow ipv4 original-input
  - netflow ipv4 original-output
  - netflow ipv4 protocol-port

  For more information about Flow Records, see the "Flow Record Definition" section on page 11-2

SUMMARY STEPS

1. **config t**
2. **flow monitor** *name*
3. **description** *string*
4. **exporter** *name*
5. **record** *name*
6. **timeout** {**active** *value* | **inactive** *value*}

7. **cache** {**size** *value*}

8. **show flow monitor** [*name*]

9. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**config t** **Example:** n1000v# config t n1000v(config)#	Places you in the CLI Global Configuration mode.
Step 2	**flow monitor** *name* **Example:** n1000v(config)# flow monitor MonitorTest n1000v(config-flow-monitor)#	Creates a flow monitor, by name, saves it in the running configuration, and then places you in the CLI Flow Monitor Configuration mode,
Step 3	**description** *string* **Example:** n1000v(config-flow-monitor)# description Ipv4Monitor	(Optional) For the specified flow monitor, adds a descriptive string, of up to 63 alphanumeric characters, and saves it in the running configuration.
Step 4	**exporter** *name* **Example:** n1000v(config-flow-monitor)# exporter Exportv9	For the specified flow monitor, adds an existing flow exporter and saves it in the running configuration.
Step 5	**record** {*name* | **netflow** {**ipv4**}} **Example using Cisco Nexus 1000V pre-defined record:** n1000v(config-flow-monitor)# record netflow-original **Example using user-defined record:** n1000v(config-flow-monitor)# record RecordTest	For the specified flow monitor, adds an existing flow record and saves it in the running configuration. • **name**: The name of a flow record you have previously created, or the name of a Cisco provided pre-defined flow record. • **netflow**: Traditional NetFlow collection schemes – **ipv4**: Traditional IPv4 NetFlow collection schemes
Step 6	**timeout** {**active** *value* | **inactive** *value*} **Example:** n1000v(config-flow-monitor)# timeout inactive 600	(Optional) For the specified flow monitor, specifies an aging timer and its value for aging entries from the cache, and saves them in the running configuration. • **active**: Active, or long, timeout. Allowable values are from 60 to 4092 seconds. Default is 1800. • **inactive**: Inactive or normal timeout. Allowable values are from 15 to 4092 seconds. Default is 15.
Step 7	**cache** {**size** *value*} **Example:** n1000v(config-flow-monitor)# cache size 15000	(Optional) For the specified flow monitor, specifies the cache size, from 256 to 16384, entries, and saves it in the running configuration. Default is 4096. **Note** This option is used to limit the impact of the monitor cache on memory and performance.

	Command	Purpose

Step 8	**show flow monitor** [*name*]
	
	Example:
	`n1000v(config-flow-monitor)# show flow`
	`monitor Monitor Test`
		(Optional) Displays information about existing flow monitors.

Step 9	**copy running-config startup-config**
	
	Example:
	`n1000v(config-flow-monitor)# copy`
	`running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

The following is an example of creating a flow exporter:

```
n1000v(config)# flow monitor MonitorTest
n1000v(config-flow-monitor)# description Ipv4Monitor
n1000v(config-flow-monitor)# exporter ExportTest
n1000v(config-flow-monitor)# record RecordTest
n1000v(config-flow-monitor)# cache size 15000
n1000v(config-flow-monitor)# timeout inactive 600
n1000v(config-flow-monitor)# show flow monitor MonitorTest
Flow Monitor monitortest:
    Use count: 0
    Inactive timeout: 600
    Active timeout: 1800
    Cache Size: 15000
n1000v(config-flow-monitor)#
```

Assigning a Flow Monitor to an Interface

Use this procedure to assign a flow monitor to an interface.

BEFORE YOU BEGIN

- You know the name of the flow monitor you want to use for the interface.
- You know the interface type and its number.

SUMMARY STEPS

1. **config t**
2. **interface** *interface-type interface-number*
3. **ip flow monitor** *name* {**input** | **output**}
4. **show flow** *interface-type interface-number*
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	**config t**
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you in the CLI Global Configuration mode.

Step 2	**interface** *interface-type interface-number*
	
	Example:
	`n1000v(config)# interface veth 2`
	`n1000v(config-if)#`
		Places you in the CLI Interface Configuration mode for the specified interface.

Step 3	**ip flow monitor** *name* {**input**	**output**}
		
		Example:
		`n1000v(config-if)# ip flow monitor`
		`MonitorTest output`
			For the specified interface, assigns a flow monitor for input or output packets and saves it in the running configuration.

Step 4	**show flow** *interface-type interface-number*
	
	Example:
	`n1000v(config-if# show flow interface`
	`veth 2`
		(Optional) For the specified interface, displays the NetFlow configuration.

Step 5	**copy running-config startup-config**
	
	Example:
	`n1000v(config-if)# copy running-config`
	`startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

The following is an example showing how to assign a flow monitor to an interface:

```
n1000v(config)# interface veth 2
n1000v(config-if)# ip flow monitor MonitorTest output
n1000v(config-if)# show flow interface veth 2
Interface veth 2:
    Monitor: MonitorTest
    Direction: Output
n1000v(config-if)#
```

Verifying the NetFlow Configuration

To verify the NetFlow configuration, use the commands in Table 11-2:

Table 11-2   Verifying the NetFlow Configuration

Command	Purpose
show flow exporter [*name*]	Displays information about NetFlow flow exporter maps. See Example 11-5 on page 11-19.
show flow interface [*interface-type number*]	Displays information about NetFlow interfaces. See Example 11-6 on page 11-20.

Table 11-2   Verifying the NetFlow Configuration (continued)

Command	Purpose
show flow monitor [*name* [**cache module** *number* |**statistics module** *number*]]	Displays information about NetFlow flow monitors. **Note** The **show flow monitor cache** command differs from the **show flow monitor statistics** command in that the cache command also displays cache entries. Since each processor has its own cache, all output of these commands is based on the number of processors on the server (also called module or host). When more than one processor is involved in processing packets for a single flow, then the same flow appears for each processor. See the following examples: • Example 11-7Show flow monitor, page 11-20 • Example 11-8Show flow monitor cache module, page 11-20 • Example 11-9Show flow monitor statistics module, page 11-21
show flow record [*name*]	Displays information about NetFlow flow records.

Example 11-5   Show flow exporter

```
n1000v(config-flow-exporter-version-9)# show flow exporter ExportTest
Flow exporter ExportTest:
    Description: ExportHamilton
    Destination: 192.0.2.1
    VRF: default (1)
    Destination UDP Port 200
    Source Interface 2
    DSCP 2
    Export Version 9
        Exporter-stats timeout 1200 seconds
        Data template timeout 1200 seconds
    Exporter Statistics
        Number of Flow Records Exported 0
        Number of Templates Exported 0
        Number of Export Packets Sent 0
        Number of Export Bytes Sent 0
        Number of Destination Unreachable Events 0
        Number of No Buffer Events 0
        Number of Packets Dropped (No Route to Host) 0
        Number of Packets Dropped (other) 0
        Number of Packets Dropped (LC to RP Error) 0
        Number of Packets Dropped (Output Drops) 1
        Time statistics were last cleared: Never
n1000v(config-flow-exporter-version-9)#
```

Example 11-6   Show flow interface

```
n1000v(config-if)# show flow interface VEth2
Interface veth2:
     Monitor: MonitorTest
     Direction: Output
n1000v(config-if)#
```

Example 11-7   Show flow monitor

```
n1000v(config)# show flow monitor
Flow Monitor MonitorTest:
     Description: Ipv4Monitor
     Use count: 1
     Flow Record: test
     Flow Exporter: ExportTest
     Inactive timeout: 15
     Active timeout: 1800
     Cache Size: 15000
Flow Monitor MonitorIpv4:
     Description: exit
     Use count: 70
     Flow Record: RecordTest
     Flow Exporter: ExportIpv4
     Inactive timeout: 15
     Active timeout: 1800
     Cache Size: 4096
n1000v(config)#
```

Example 11-8   Show flow monitor cache module

```
n1000v# show flow monitor test_mon cache module 5
Cache type:                       Normal
Cache size (per-processor):       4096
High Watermark:                   2
Flows added:                      102
Flows aged:                       099
    - Active timeout              0
    - Inactive timeout            099
    - Event aged                  0
    - Watermark aged              0
    - Emergency aged              0
    - Permanent                   0
    - Immediate aged              0
    - Fast aged                   0

Cache entries on Processor0
    - Active Flows:               2
    - Free Flows:                 4094

  IPV4 SRC ADDR    IPV4 DST ADDR  IP PROT        INTF INPUT           INTF OUTPUT    FLOW DIRN
===============  ===============  =======  ====================  ====================  =========
       0.0.0.0  255.255.255.255  17                    Veth1                            Input
  7.192.192.10      7.192.192.2  1                     Veth1                 Eth5/2    Input

Cache entries on Processor1
    - Active Flows:               0
    - Free Flows:                 4096

Cache entries on Processor2
    - Active Flows:               1
    - Free Flows:                 4095
```

```
  IPV4 SRC ADDR     IPV4 DST ADDR  IP PROT            INTF INPUT            INTF OUTPUT  FLOW DIRN
  ==============    ==============  =======  ====================  ====================  =========
    7.192.192.10      7.192.192.1  1                        Veth1                Eth5/2     Input

Cache entries on Processor3
   - Active Flows:               0
   - Free Flows:              4096

Cache entries on Processor4
   - Active Flows:               0
   - Free Flows:              4096

Cache entries on Processor5
   - Active Flows:               0
   - Free Flows:              4096

Cache entries on Processor6
   - Active Flows:               0
   - Free Flows:              4096

Cache entries on Processor7
   - Active Flows:               0
   - Free Flows:              4096
```

Example 11-9   Show flow monitor statistics module

```
NX-1000v# show flow monitor test_mon statistics module 5
Cache type:                     Normal
Cache size (per-processor):     4096
High Watermark:                 2
Flows added:                    105
Flows aged:                     103
   - Active timeout             0
   - Inactive timeout           103
   - Event aged                 0
   - Watermark aged             0
   - Emergency aged             0
   - Permanent                  0
   - Immediate aged             0
   - Fast aged                  0

Cache entries on Processor0
   - Active Flows:               0
   - Free Flows:              4096

Cache entries on Processor1
   - Active Flows:               1
   - Free Flows:              4095

Cache entries on Processor2
   - Active Flows:               1
   - Free Flows:              4095

Cache entries on Processor3
   - Active Flows:               0
   - Free Flows:              4096

Cache entries on Processor4
   - Active Flows:               0
   - Free Flows:              4096

Cache entries on Processor5
```

```
       - Active Flows:                   0
       - Free Flows:                  4096

Cache entries on Processor6
       - Active Flows:                   0
       - Free Flows:                  4096

Cache entries on Processor7
       - Active Flows:                   0
       - Free Flows:                  4096
```

Example 11-10 Show flow record

```
n1000v(config-flow-record)# show flow record RecordTest
Flow record RecordTest:
    Description: Ipv4flow
    No. of users: 0
    Template ID: 0
    Fields:
        match ipv4 destination address
        match interface input
        match interface output
        match flow direction
        collect counter packets
n1000v(config-flow-record)#
```

Configuration Example for NetFlow

The following example shows how to configure a flow monitor using a new flow record and applying it to an interface.

```
n1000v# config t
n1000v(config)# flow record RecordTest
n1000v(config-flow-record)# description Ipv4flow
n1000v(config-flow-record)# match ipv4 destination address
n1000v(config-flow-record)# collect counter packets
n1000v(config-flow-record)# exit
n1000v(config)# flow exporter ExportTest
n1000v(config-flow-exporter)# description ExportHamilton
n1000v(config-flow-exporter)# destination 192.0.2.1
n1000v(config-flow-exporter)# dscp 2
n1000v(config-flow-exporter)# source mgmt 0
n1000v(config-flow-exporter)# transport udp 200
n1000v(config-flow-exporter)# version 9
n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200
n1000v(config-flow-exporter-version-9)# template data timeout 1200
n1000v(config-flow-exporter-version-9)# exit
n1000v(config-flow-exporter)# exit
n1000v(config)# flow monitor MonitorTest
n1000v(config-flow-monitor)# description Ipv4Monitor
n1000v(config-flow-monitor)# exporter ExportTest
n1000v(config-flow-monitor)# record RecordTest
n1000v(config-flow-monitor)# exit
n1000v(config)# interface veth 2/1
n1000v(config-if)# ip flow monitor MonitorTest output
n1000v(config-if)# show flow interface veth 2
Interface veth 2:
    Monitor: MonitorTest
    Direction: Output
n1000v(config-if)#
```

The following example shows how to configure flow monitor using a predefined record and applying it to an interface.

```
n1000v# config t
n1000v(config)# flow exporter ExportTest
n1000v(config-flow-exporter)# description ExportHamilton
n1000v(config-flow-exporter)# destination 192.0.2.1
n1000v(config-flow-exporter)# dscp 2
n1000v(config-flow-exporter)# source mgmt 0
n1000v(config-flow-exporter)# transport udp 200
n1000v(config-flow-exporter)# version 9
n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200
n1000v(config-flow-exporter-version-9)# template data timeout 1200
n1000v(config-flow-exporter-version-9)# exit
n1000v(config-flow-exporter)# exit
n1000v(config)# flow monitor MonitorTest
n1000v(config-flow-monitor)# description Ipv4Monitor
n1000v(config-flow-monitor)# exporter ExportTest
n1000v(config-flow-monitor)# record netflow-original
n1000v(config-flow-monitor)# exit
n1000v(config)# interface veth 2/1
n1000v(config-if)# ip flow monitor MonitorTest output
n1000v(config-if)# show flow interface veth 2
Interface veth 2:
    Monitor: MonitorTest
    Direction: Output
n1000v(config-if)#
```

Additional References

For additional information related to implementing NetFlow, see the following sections:

- Related Documents, page 11-23
- Standards, page 11-24

Related Documents

Related Topic	Document Title
Cisco NetFlow Overview	http://cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html
Complete command syntax, command mode, command history, defaults, usage guidelines, and examples for all Cisco Nexus 1000V commands.	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(3)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for NetFlow

This section provides the NetFlow feature release history.

Feature Name	Releases	Feature Information
NAM support for NetFlow data sources	4.0(4)SV1(3)	NAM support for NetFlow data sources was added.
NetFlow	4.0(4)SV1(1)	NetFlow was introduced.

C H A P T E R 12

Configuring System Message Logging

This chapter describes how to configure system message logging.

Information About System Message Logging

You can use system message logging to control the destination and to filter the severity level of messages that system processes generate. You can configure logging to terminal sessions, a log file, and syslog servers on remote systems.

System message logging is based on RFC 3164. For more information about the system message format and the messages that the device generates, see the *Cisco NX-OS System Messages Reference*.

By default, the device outputs messages to terminal sessions. For information about configuring logging to terminal sessions, see the .

describes the severity levels used in system messages. When you configure the severity level, the system outputs messages at that level and lower.

.

Table 12-1　　System Message Severity Levels

Level	Description
0 – emergency	System unusable
1 – alert	Immediate action needed
2 – critical	Critical condition
3 – error	Error condition
4 – warning	Warning condition
5 – notification	Normal but significant condition
6 – informational	Informational message only
7 – debugging	Appears during debugging only

The device logs the most recent 100 messages of severity 0, 1, or 2.

You can configure which system messages should be logged based on the facility that generated the message and its severity level. For information about facilities, see the . For information about configuring the severity level by module and facility, see the .

syslog servers run on remote systems that are configured to log system messages based on the syslog protocol. You can configure up to three syslog servers. For information about configuring syslog servers, see the "Configuring syslog Servers" section on page 12-11.

Note   When the device first initializes, messages are sent to syslog servers only after the network is initialized.

System Message Logging Facilities

Table 12-2 lists the facilities that you can use in system message logging configuration.

Table 12-2   ***System Message Logging Facilities***

Facility	Description
aaa	AAA manager
aclmgr	ACL manager
adjmgr	Adjacency Manager
all	Keyword that represents all facilities
arbiter	Arbiter manager
arp	ARP manager
auth	Authorization system
authpriv	Private authorization system
bootvar	Bootvar
callhome	Call home manager
capability	MIG utilities daemon
cdp	CDP manager
cert-enroll	Certificate enroll daemon
cfs	CFS manager
clis	CLIS manager
cmpproxy	CMP proxy manager
copp	CoPP manager
core	Core daemon
cron	Cron and at scheduling service
daemon	System daemons
dhcp	DHCP manager
diagclient	GOLD diagnostic client manager
diagmgr	GOLD diagnostic manager
eltm	ELTM manager
ethpm	Ethernet PM manager
evmc	EVMC manager
evms	EVMS manager

Table 12-2   System Message Logging Facilities (continued)

Facility	Description
feature-mgr	Feature manager
fs-daemon	Fs daemon
ftp	File transfer system
glbp	GLBP manager
hsrp	HSRP manager
im	IM manager
ipconf	IP configuration manager
ipfib	IP FIB manager
kernel	OS kernel
l2fm	L2 FM manager
l2nac	L2 NAC manager
l3vm	L3 VM manager
license	Licensing manager
local0	Local use daemon
local1	Local use daemon
local2	Local use daemon
local3	Local use daemon
local4	Local use daemon
local5	Local use daemon
local6	Local use daemon
local7	Local use daemon
lpr	Line printer system
m6rib	M6RIB manager
mail	Mail system
mfdm	MFDM manager
module	Module manager
monitor	Ethernet SPAN manager
mrib	MRIB manager
mvsh	MVSH manager
news	USENET news
nf	NF manager
ntp	NTP manag
otm	GLBP manager
pblr	PBLR manager
pfstat	PFSTAT manager
pixm	PIXM manager

Table 12-2   System Message Logging Facilities (continued)

Facility	Description
pixmc	PIXMC manager
pktmgr	Packet manager
platform	Platform manager
pltfm_config	PLTFM configuration manager
plugin	Plug-in manager
port-channel	Port channel manager
port_client	Port client manager
port_lb	Diagnostic port loopback test manager
qengine	Q engine manager
radius	RADIUS manager
res_mgr	Resource? manager
rpm	RPM manager
security	Security manager
session	Session manager
spanning-tree	Spanning tree manager
syslog	Internal syslog manager
sysmgr	System manager
tcpudp	TCP and UDP manager
u2	U2 manager
u6rib	U6RIB manager
ufdm	UFDM manager
urib	URIB manager
user	User process
uucp	Unix-to-Unix copy system
vdc_mgr	VDC manager
vlan_mgr	VLAN manager
vmm	VMM manager
vshd	VSHD manager
xbar	XBAR manager
xbar_client	XBAR client manager
xbar_driver	XBAR driver manager
xml	XML agent

Guidelines and Limitations

System messages are logged to the console and the logfile by default.

Default Settings

Table 12-3 lists the default settings for system message logging.

Table 12-3   *System Message Logging Defaults*

Parameter	Default
Console logging	Enabled at severity level 2
Monitor logging	Enabled at severity level 5
Log file logging	Enabled to log messages at severity level 5
Module logging	Enabled at severity level 5
Facility logging	Enabled; for severity levels, see the "System Message Logging Facilities" section on page 12-2.
Time-stamp units	Seconds
syslog server logging	Disabled
syslog server configuration distribution	Disabled

Configuring System Message Logging

This section includes the following topics:

Note   Be aware that NX-OS commands may differ from the Cisco IOS commands.

Configuring System Message Logging to Terminal Sessions

Use this procedure to log messages by severity level to console, telnet, and SSH sessions.

BEFORE YOU BEGIN

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

- By default, logging is enabled for terminal sessions.

SUMMARY STEPS

1. **terminal monitor**
2. **config t**
3. **logging console** [*severity-level*]
4. **show logging console**
5. **logging monitor** [*severity-level*]
6. **show logging monitor**
7. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`terminal monitor`
	
	Example:
	`n1000v# terminal monitor`
	`n1000v#`
		Enables the device to log messages to the console.

Step 2	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Enters global configuration mode.

Step 3	`logging console` [*severity-level*]
	
	Example:
	`n1000v(config)# logging console 2`
	`n1000v(config)#`
		Configures the device to log messages to the console session based on a specified severity level or higher. Severity levels, which can range from 0 to 7, are listed in Table 12-1. If the severity level is not specified, the default of 2 is used.

Step 4	`show logging console`	(Optional) Displays the console logging configuration.

Step 5	`logging monitor` [*severity-level*]
	
	Example:
	`n1000v(config)# logging monitor 3`
	`n1000v(config)#`
		Enables the device to log messages to the monitor based on a specified severity level or higher. The configuration applies to telnet and SSH sessions. Severity levels, which can range from 0 to 7, are listed in Table 12-1. If the severity level is not specified, the default of 2 is used.

Step 6	`show logging monitor`	(Optional) Displays the monitor logging configuration.

Command	Purpose

Step 7   `copy running-config startup-config`
 	
Example:
`n1000v(config)# copy running-config`
`startup-config`
	(Optional) Copies the running configuration to the startup configuration.

Example:
```
n1000v# terminal monitor
n1000v# config t
n1000v(config)# logging console 2
n1000v(config)# show logging console
Logging console:                 enabled (Severity: critical)
n1000v(config)# logging monitor 3
n1000v(config)# show logging monitor
Logging monitor:                 enabled (Severity: errors)
n1000v(config)#
n1000v(config)# copy running-config startup-config
```

Restoring System Message Logging Defaults for Terminal Sessions

Use the following commands in the CLI Global Configuration mode to restore default settings for system message logging for terminal sessions.

Command	Description

no logging console [*severity-level*]
 	
Example:
`n1000v(config)# no logging console`
`n1000v(config)#`
	Disables the device from logging messages to the console.

no logging monitor [*severity-level*]
 	
Example:
`n1000v(config)# no logging monitor 3`
`n1000v(config)#`
	Disables logging messages to telnet and SSH sessions.

Configuring System Message Logging for Modules

Use this procedure to configure the severity level and time-stamp units of messages logged by modules.

SUMMARY STEPS

1. **config t**

2. **logging module** [*severity-level*]

3. **show logging module**

4. **logging timestamp** {**microseconds** | **milliseconds** | **seconds**}

5. **show logging timestamp**

6. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

Step 2	`logging module` [*severity-level*] **Example:** `n1000v(config)# logging module 3`	Enables module log messages that have the specified severity level or higher. Severity levels, which range from 0 to 7, are listed in Table 12-1. If the severity level is not specified, the default of 5 is used.
Step 3	`show logging module`	(Optional) Displays the module logging configuration.

Step 4	`logging timestamp {microseconds | milliseconds | seconds}`
	
	Example:
	`n1000v(config)# logging timestamp microseconds`
		Sets the logging time-stamp units. The default unit is seconds.

Step 5	`show logging timestamp`	(Optional) Displays the logging time-stamp units configured.

Step 6	`copy running-config startup-config`
	
	Example:
	`n1000v(config)# copy running-config startup-config`
		(Optional) Copies the running configuration to the startup configuration.

The following example shows how to configure system message logging for modules.

```
n1000v# config t
n1000v(config)# logging module 3

n1000v(config)# show logging module
Logging linecard:                enabled (Severity: errors)
n1000v(config)# logging timestamp microseconds
n1000v(config)# show logging timestamp
Logging timestamp:               Microseconds
n1000v(config)# copy running-config
```

Restoring System Message Logging Defaults for Modules

Use the following commands in the CLI Global Configuration mode to restore default settings for system message logging for modules.

Command	Description

`no logging module` [*severity-level*]
 	
Example:
`n1000v(config)# no logging module 3`
`n1000v(config)#`
	Restores the default severity level for logging module system messages.

`no logging timestamp` {`microseconds` | `milliseconds` | `seconds`}
 	
Example:
`n1000v(config)# no logging timestamp`
`milliseconds`
	Resets the logging time-stamp unit to the default (seconds).

Configuring System Message Logging for Facilities

Use this procedure to configure the severity level and time-stamp units of messages logged by facilities.

SUMMARY STEPS

1. **config t**
2. **logging level** *facility severity-level*
3. **show logging level** [*facility*]
4. **logging timestamp** {**microseconds** | **milliseconds** | **seconds**}
5. **show logging timestamp**
6. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Places you into CLI Global Configuration mode.

	Command	Purpose
Step 2	**logging level** *facility severity-level* **Example:** n1000v(config)# logging level aaa 3 n1000v(config)#	Enables logging messages from the specified facility that have the specified severity level or higher. The facilities are listed in the "System Message Logging Facilities" section on page 12-2. Severity levels, which range from 0 to 7, are listed in Table 12-1. To apply the same severity level to all facilities, use the **all** facility. For defaults, see the **show logging level** command.
Step 3	**show logging level** [*facility*] **Example:** n1000v(config)# show logging level aaa	(Optional) Displays the logging level configuration and the system default level by facility. If you do not specify a facility, the device displays levels for all facilities.
Step 4	**logging timestamp** {**microseconds** | **milliseconds** | **seconds**} **Example:** n1000v(config)# logging timestamp microseconds	Sets the logging time-stamp units. The default unit is seconds.
Step 5	**show logging timestamp**	(Optional) Displays the logging time-stamp units configured.
Step 6	**copy running-config startup-config** **Example:** n1000v(config)# copy running-config startup-config	(Optional) Copies the running configuration to the startup configuration.

The following example shows how to configure system message logging for facilities.

```
n1000v# config t
n1000v(config)# logging level aaa 3
n1000v(config)# show logging level aaa
Facility          Default Severity        Current Session Severity
--------          ----------------        ------------------------
aaa                     2                           3

0(emergencies)          1(alerts)       2(critical)
3(errors)               4(warnings)     5(notifications)
6(information)          7(debugging)
logging timestamp microseconds
n1000v(config)# show logging timestamp
Logging timestamp:              Microseconds
copy running-config startup-config
```

Restoring System Message Logging Defaults for Facilities

Use the following commands to restore system message logging defaults for facilities.

Command	Description

`no logging level` [*facility severity-level*]
 	
Example:
`n1000v(config)# no logging level aaa 3`
`n1000v(config)#`
	Restores the default logging severity level for the specified facility. If you do not specify a facility and severity level, the device resets all facilities to their default levels.

`no logging timestamp {microseconds |` `milliseconds	seconds}`  **Example:** `n1000v(config)# no logging timestamp` `milliseconds`	Resets the logging time-stamp unit to the default (seconds).

Configuring syslog Servers

Use this procedure to configure syslog servers for system message logging.

SUMMARY STEPS

1. **config t**
2. **logging server** *host* [*severity-level* [**use_vrf** *vrf-name*]]
3. **show logging server**
4. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose

Step 1	`config t`
	
	Example:
	`n1000v# config t`
	`n1000v(config)#`
		Enters global configuration mode.

Step 2	`logging server` *host* [*severity-level* [**use-vrf** *vrf-name*]] **Example:** `n1000v(config)# logging server 10.10.2.2 7`	Configures a syslog server at the specified host name or IPv4 or IPv6 address. You can limit logging of messages to a particular VRF by using the **use_vrf** keyword. Severity levels, which range from 0 to 7, are listed in Table 12-1. The default outgoing facility is local7. The example forwards all messages on facility local 7.

Step 3	`show logging server`
	
	Example:
	`n1000v(config)# show logging server`
	`Logging server:             enabled`
	`{10.10.2.2}`
	`        server severity:      debugging`
	`        server facility:     local7`
		(Optional) Displays the syslog server configuration.

Step 4	`copy running-config startup-config`
	
	Example:
	`n1000v(config)# copy running-config`
	`startup-config`
		(Optional) Copies the running configuration to the startup configuration.

Restoring System Message Logging Defaults for Servers

Use the following command to restore server system message logging default.

Command	Description
`no logging server` *host* **Example:** `n1000v(config)# no logging server host`	Removes the logging server for the specified host.

Using a UNIX or Linux System to Configure Logging

Use this procedure on a UNIX or Linux system to configure message logging.

BEFORE YOU BEGIN

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

- The following are the UNIX or Linux fields to configure for syslog:

.

Field	Description
Facility	Creator of the message, which can be auth, authpriv, cron, daemon, kern, lpr, mail, mark, news, syslog, user, local0 through local7, or an asterisk (*) for all. These facility designators allow you to control the destination of messages based on their origin. **Note** Check your configuration before using a local facility.
Level	Minimum severity level at which messages are logged, which can be debug, info, notice, warning, err, crit, alert, emerg, or an asterisk (*) for all. You can use none to disable a facility.
Action	Destination for messages, which can be a filename, a host name preceded by the at sign (@), or a comma-separated list of users or an asterisk (*) for all logged-in users.

DETAILED STEPS

Step 1   On the UNIX or Linux system, add the following line to the file, /var/log/myfile.log:

facility.level <five tab characters> *action*

Example:
debug.local7   /var/log/myfile.log

Step 2   Create the log file by entering these commands at the shell prompt:

```
$ touch /var/log/myfile.log
$ chmod 666 /var/log/myfile.log
```

Step 3   Make sure the system message logging daemon reads the new changes by checking myfile.log after entering this command:

```
$ kill -HUP ~cat /etc/syslog.pid~
```

Displaying Log Files

Use this procedure to display messages in the log file.

SUMMARY STEPS

1. **show logging last** *number-lines*

DETAILED STEPS

	Command	Purpose
Step 1	`show logging last number-lines`	Displays the last number of lines in the logging file. You can specify from 1 to 9999 for the last number of lines.

The following example shows the last five lines in the logging file.

```
n1000v#  show logging last 5
2008 Aug 31 09:37:04 CP-beta2 %KERN-3-SYSTEM_MSG: packet_recvms
g: truncated packet (size=1514 left=1500) - kernel
2008 Aug 31 09:37:04 CP-beta2 %KERN-3-SYSTEM_MSG: packet_recvms
g: truncated packet (size=1514 left=1500) - kernel
2008 Aug 31 09:37:05 CP-beta2 %KERN-3-SYSTEM_MSG: packet_recvms
g: truncated packet (size=1514 left=1500) - kernel
2008 Aug 31 09:37:05 CP-beta2 %KERN-3-SYSTEM_MSG: packet_recvms
g: truncated packet (size=1514 left=1500) - kernel
2008 Aug 31 09:37:05 CP-beta2 %KERN-3-SYSTEM_MSG: packet_recvms
g: truncated packet (size=1514 left=1500) - kernel
n1000v#
```

Verifying the System Message Logging Configuration

To verify the system message logging configuration, use one of the following commands:

Command	Purpose
show logging console	Displays the console logging configuration. See Example 12-1 on page 12-15
show logging info	Displays the logging configuration. See Example 12-2 on page 12-15
show logging last *number-lines*	Displays the last number of lines of the log file. See Example 12-3 on page 12-16
show logging level [*facility*]	Displays the facility logging severity level configuration. See Example 12-4 on page 12-16
show logging module	Displays the module logging configuration. See Example 12-5 on page 12-17
show logging monitor	Displays the monitor logging configuration. See Example 12-6 on page 12-17
show logging server	Displays the syslog server configuration. See Example 12-7 on page 12-17
show logging session	Displays the logging session status. See Example 12-8 on page 12-17

Command	Purpose
show logging status	Displays the logging status.
	See Example 12-9 on page 12-17
show logging timestamp	Displays the logging time-stamp units configuration.
	See Example 12-10 on page 12-17

Example 12-1 `show logging console`

```
n1000v# show logging console
Logging console:               disabled
n1000v#
```

Example 12-2 `show logging info`

```
n1000v# show logging info

Logging console:               enabled (Severity: critical)
Logging monitor:               enabled (Severity: notifications)
Logging linecard:              enabled (Severity: notifications)
Logging timestamp:             Seconds
Logging server:                disabled
Logging logfile:               enabled
        Name - g/external/messages: Severity - notifications Size - 4194304

Facility          Default Severity          Current Session Severity
--------          ----------------          -----------------------
aaa                      2                            2
auth                     0                            0
authpriv                 3                            3
bootvar                  5                            5
callhome                 2                            2
cdp                      2                            2
cert_enroll              2                            2
cfs                      3                            3
confcheck                2                            2
cron                     3                            3
daemon                   3                            3
diagclient               2                            2
diagmgr                  2                            2
eth_port_channel         5                            5
ethpm                    5                            5
evmc                     5                            5
evms                     2                            2
feature-mgr              2                            2
ftp                      3                            3
ifmgr                    5                            5
igmp_1                   3                            3
ip                       2                            2
ipv6                     2                            2
kern                     6                            6
l2fm                     2                            2
licmgr                   6                            6
local0                   3                            3
local1                   3                            3
local2                   3                            3
local3                   3                            3
local4                   3                            3
```

```
local5                  3                       3
local6                  3                       3
local7                  3                       3
lpr                     3                       3
mail                    3                       3
mfdm                    2                       2
module                  5                       5
monitor                 7                       7
msp                     2                       2
mvsh                    2                       2
news                    3                       3
ntp                     2                       2
otm                     3                       3
pblr                    2                       2
pixm                    2                       2
pixmc                   2                       2
platform                5                       5
portprofile             5                       5
private-vlan            3                       3
radius                  2                       2
res_mgr                 2                       2
rpm                     2                       2
sal                     2                       2
securityd               2                       2
sksd                    3                       3
stp                     3                       3
syslog                  3                       3
sysmgr                  3                       3
ufdm                    2                       2
urib                    3                       3
user                    3                       3
uucp                    3                       3
vdc_mgr                 6                       6
vim                     5                       5
vlan_mgr                2                       2
vms                     5                       5
vshd                    5                       5
xmlma                   3                       3

0(emergencies)          1(alerts)       2(critical)
3(errors)               4(warnings)     5(notifications)
6(information)          7(debugging)
n1000v$
```

Example 12-3 `show logging last`

```
n1000v# show logging last 5
2008 Jul 29 17:52:42 S22-DCOS %ETHPORT-5-IF_UP: Interface Ethernet2/5 is up in mode access
2008 Jul 29 17:52:43 S22-DCOS %ETHPORT-5-IF_UP: Interface Ethernet2/2 is up in mode trunk
2008 Jul 29 17:52:43 S22-DCOS %ETHPORT-5-IF_UP: Interface Ethernet2/4 is up in mode access
2008 Jul 29 17:53:04 S22-DCOS %SYSMGR-3-BASIC_TRACE: process_cfg_write: PID 1858 with
message rcvd cfg_action from
sap 0x545 for vdc 1 at time 1217353984 .
2008 Jul 29 17:53:04 S22-DCOS clis[2558]: CLI-3-NVDB: Batched send failed for component:
clic
n1000v#
```

Example 12-4 show logging level aaa

```
n1000v# show logging level aaa
Facility        Default Severity        Current Session Severity
--------        ----------------        ------------------------
aaa                     2                       2
```

```
0(emergencies)         1(alerts)       2(critical)
3(errors)              4(warnings)     5(notifications)
6(information)         7(debugging)
n1000v#
```

Example 12-5 show logging module

```
n1000v# show logging module
Logging linecard:              enabled (Severity: notifications)
n1000v#
```

Example 12-6 show logging monitor

```
n1000v# show logging monitor
Logging monitor:               enabled (Severity: errors)
n1000v#
```

Example 12-7 show logging server

```
n1000v# show logging server
Logging server:                enabled
{10.10.2.2}
        server severity:       debugging
        server facility:       local7
n1000v#
```

Example 12-8 show logging session status

```
n1000v# show logging session status
Last Action Time Stamp    : Fri Nov 18 11:28:55 1910
Last Action               : Distribution Enable
Last Action Result        : Success
Last Action Failure Reason : none
n1000v#
```

Example 12-9 show logging status

```
n1000v# show logging status
Fabric Distribute    : Enabled
Session State        : IDLE
n1000v#
```

Example 12-10 show logging timestamp

```
n1000v# show logging timestamp
Logging timestamp:             Seconds
n1000v#
```

System Message Logging Example Configuration

The following example shows how to configure system message logging:

```
config t
  logging console 3
  logging monitor 3
  logging logfile my_log 6
  logging module 3
  logging level aaa 2
  logging timestamp milliseconds
  logging distribute
  logging server 172.28.254.253
  logging server 172.28.254.254 5 local3
  logging commit
  copy running-config startup-config
```

Additional References

For additional information related to implementing system message logging, see the following sections:

- Related Documents, page 12-18
- Standards, page 12-18

Related Documents

Related Topic	Document Title
System messages	*Cisco NX-OS System Messages Reference*
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(3)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for System Message Logging

This section provides the system message logging feature release history.

Feature Name	Releases	Feature Information
System Message Logging	4.0(4)SV1(1)	This feature was introduced.

CHAPTER **13**

Configuring iSCSI Multipath

This chapter describes how to configure iSCSI multipath for multiple routes between a server and its storage devices.

iSCSI Multipath Description

The iSCSI multipath feature sets up multiple routes between a server and its storage devices for maintaining a constant connection and balancing the traffic load. The multipathing software handles all input and output requests and passes them through on the best possible path. Traffic from host servers is transported to shared storage using the iSCSI protocol that packages SCSI commands into iSCSI packets and transmits them on the Ethernet network.

iSCSI multipath provides failover. In the event a path, or any of its components, fails, the server selects another available path. In addition to path failover, multipathing load balances by distributing storage loads across multiple physical paths to reduce or remove potential bottlenecks.

The following are descriptions of the iSCSI multipath processes:

iSCSI Multipath Process	Provided by Cisco Nexus 1000V	Provided by VMware code in VMkernel	Description
Uplink Pinning	X		Each VMKernel port created for iSCSI access are pinned to one physical NIC. This overrides any NIC teaming policy or port bundling policy. All traffic from the VMKernel port uses only the pinned uplink to reach the upstream switch.
Storage Binding		X	The ESX/ESXi host creates iSCSI adapters, also called VMware iSCSI host bus adapter (VMHBA) adapters, for the physical NICs. For Software iSCSI, only one VMHBA is created for all the physical NICs. For Hardware iSCSI, one VMHBA is created for each physical NIC that supports iSCSI offload in hardware. Storage binding is the step in which each VMKernel port is pinned to the VMHBA associated with the physical NIC to which the VMKernel port is pinned.
Path selection		X	

iSCSI Multipath Process	Provided by Cisco Nexus 1000V	Provided by VMware code in VMkernel	Description
Path failover		X	In the event of failure, in the path or any of its components, the server selects another available path.
Load Balancing			Reduces or removes bottlenecks by distributing storage loads across multiple physical paths.

The Cisco Nexus 1000V DVS performs iSCSI multipathing regardless of the iSCSI target. The iSCSI daemon on an ESX server communicates with a target in multiple sessions by pinning two or more host VMkernel NICs on the same Cisco Nexus 1000V. Therefore, the Cisco Nexus 1000V should have a minimum of two physical NICs.

Once you enable iSCSI multipath on the port profile, using the command **capability iscsi-multipath**, the VEM automatically pins the VMkernel NICs to VM NICs.

Standard NICs connect the host to a remote iSCSI target on the IP network. The software iSCSI adapter that is built into ESX/ESXi communicates with the physical NICs through the network stack.

For detailed information about how to use VMware® ESX™ and VMware ESXi systems with an iSCSI storage area network (SAN), see the *iSCSI SAN Configuration Guide*.

iSCSI Dependent Hardware Adapter

A third-party dependent hardware iSCSI adapter, such as a Broadcom Ethernet NIC with iSCSI offload functions, offloads iSCSI and network processing from your host leveraging VMware iSCSI management and configuration interfaces.

After this NIC is installed, the dependent hardware iSCSI adapter is loaded onto the host and appears in the list of storage adapters. Although the adapter is enabled by default, to make it functional, VMkernel networking must be set up for the iSCSI traffic and the adapter must be bound to a VMkernel iSCSI port.

Only one VMHBA adapter is available for each NIC that is capable of iSCSI offload. The VMKernel port is bound to the VMHBA adapter of the hardware NIC (vmnic) to which it is pinned. In Figure 13-1, VMkernel NIC 1 is bound to vmhba33.

iSCSI Adapter on the VMware vSwitch

VMkernel iSCSI ports, for the traffic between the iSCSI adapter and the physical NIC, must be created before enabling or configuring the software or hardware iSCSI for multipathing. Only one adapter (VMHBA) is created for software iSCSI and all VMKernel ports are bound to it.

iSCSI Multipath Setup on the VMware Switch

Before enabling or configuring multipathing, networking must be configured for the software or hardware iSCSI adapter. This involves opening a VMkernel iSCSI port for the traffic between the iSCSI adapter and the physical NIC.

For software iSCSI, only one adapter is required for the entire implementation. All VMKernel ports are bound to this adapter. For example, in Figure 13-1, the following adapters and NICs are used.

Hardware iSCSI Adapter	VMkernel NIC	VM NIC
VMHBA	1	1
	2	2

For hardware iSCSI, a separate adapter is required for each NIC. Each VMkernel port is bound to the adapter of the physical VM NIC to which it is pinned. For example, in Figure 13-1, the following adapters and NICs are used.

Hardware iSCSI Adapter	VMkernel NIC	VM NIC
vmhba33	1	1
vmhba34	2	2

Figure 13-1 shows the setup of iSCSI multipath on a VMware virtual switch.

Figure 13-1   iSCSI Multipath on VMware Virtual Switch
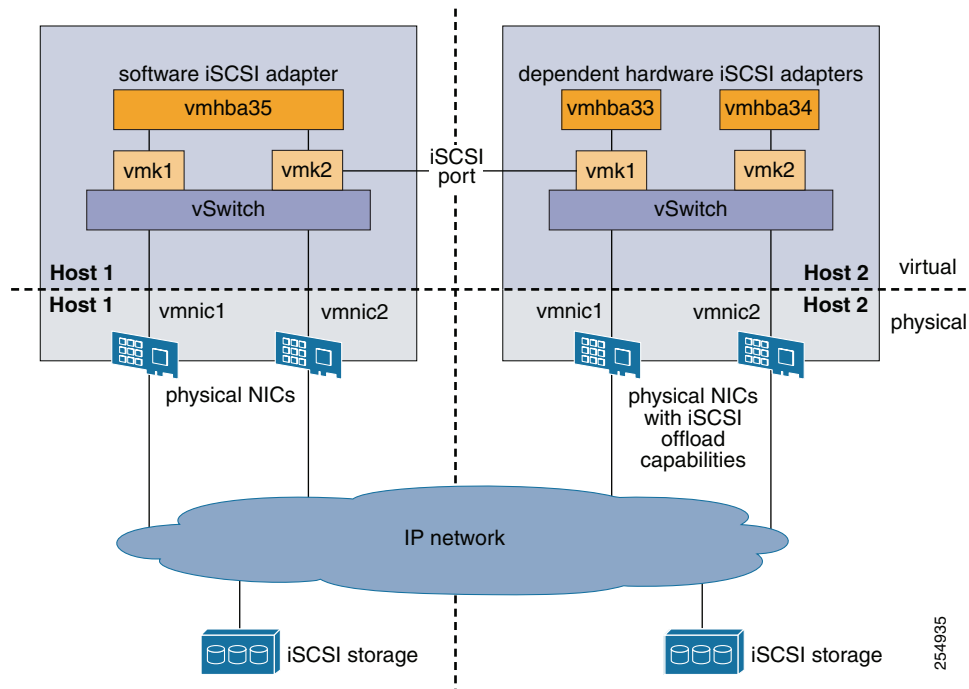
Guidelines and Limitations

The following are guidelines and limitations for the iSCSI multipath feature.

- Only port profiles of type vEthernet can be configured with **capability iscsi-multipath**.
- The port profile used for iSCSI multipath must be an access port profile. It cannot be a trunk port profile.

- The following are not allowed on a port profile configured with **capability iscsi-multipath**
  - The port profile cannot also be configured with **capability l3 control**.
  - A system VLAN change when the port profile is inherited by VMkernel NIC.
  - An access VLAN change.
  - A port mode change to trunk mode.
- Only VMkernel NIC ports can inherit a port profile configured with **capability iscsi-multipath**.
- A VMware Kernel NIC can only be pinned or assigned to one physical NIC.
- A physical NIC can have multiple VMware Kernel NICs pinned or assigned to it.
- The iSCSI initiators and storage must already be operational.
- ESX 4.0 Update1 or later supports software iSCSI multipathing only.
- ESX 4.1 or later supports both software and hardware iSCSI multipathing.
- The iSCSI adapter must be enabled and bound to a VMkernel iSCSI port.
- VMkernel networking must be functioning for the iSCSI traffic.

Prerequisites

The iSCSI Multipath feature has the following prerequisites.

- You must understand VMware iSCSI SAN storage virtualization.
- You must know how to set up the iSCSI Initiator software on your VMware ESX/ESXi host.
- The host is already functioning with the VMware ESX 4.0.1 Update 01 software release.
- You must understand iSCSI multipathing and path failover.
- A system VLAN is created on the Cisco Nexus 1000V.

  One of the uplink ports must already have this VLAN in its system VLAN range.
- The host is configured with one port channel that includes two or more physical NICs.
- VMware kernel NICs configured to access the SAN external storage are required.

Default Settings

Table 13-1 lists the default settings in the iSCSI Multipath configuration.

Table 13-1   iSCSI Multipath Defaults

Parameter	Default
Type (port-profile)	vEthernet
Description (port-profile)	None
VMware port group name (port-profile)	The name of the port profile
Switchport mode (port-profile)	Access
State (port-profile)	Disabled

Configuring a Port Profile for iSCSI Multipath

Use this section to configure communication multipathing between hosts and targets over iSCSI protocol by assigning the vEthernet interface to an iSCSI multipath port profile configured with a system VLAN.

Creating a Port Profile for VMkernel Ports

You can use this procedure to create a port profile for VMkernel ports.

BEFORE YOU BEGIN

Before starting this procedure, you must know or do the following.

- You have already configured the host with one port channel that includes two or more physical NICs.
- You have already created VMware kernel NICs to access the SAN external storage.
- A VMware Kernel NIC can only be pinned or assigned to one physical NIC.
- A physical NIC can have multiple VMware Kernel NICs pinned or assigned to it.
- Multipathing must be configured on the interface by using this procedure to create an iSCSI multipath port profile and then assigning the interface to it.
- You are logged in to the CLI in EXEC mode.
- You know the VLAN ID for the VLAN you are adding to this iSCSI multipath port profile.
    - The VLAN must already be created on the Cisco Nexus 1000V.
    - The VLAN that you assign to this iSCSI multipath port profile must be a system VLAN.
    - One of the uplink ports must already have this VLAN in its system VLAN range.
- The port profile must be an access port profile. It cannot be a trunk port profile. This procedure includes steps to configure the port profile as an access port profile.

SUMMARY STEPS

1. **config t**
2. **port-profile type vethernet***name*
3. **vmware port-group** [*name*]
4. **switchport mode access**
5. **switchport access vlan** *vlanID*
6. **no shutdown**
7. **(Optional) system vlan** *vlanID*
8. **capability iscsi-multipath**
9. **state enabled**
10. (Optional) **show port-profile** *name*
11. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	**config t** **Example:** n1000v# config t n1000v(config)#	Places you in the CLI Global Configuration mode.
Step 2	**port-profile type vethernet** *name* **Example:** n1000v(config)# port-profile type vethernet VMK-port-profile n1000v(config-port-prof)#	Places you into the CLI Port Profile Configuration mode for the specified port profile. • **type**: Defines the port-profile as Ethernet or vEthernet type. Once configured, this setting cannot be changed. The default is vEthernet type. **Note** If a port profile is configured as an Ethernet type, then it cannot be used to configure VMware virtual ports. • *name*: The port profile name can be up to 80 characters and must be unique for each port profile on the Cisco Nexus 1000V.
Step 3	**description** *profiledescription* **Example:** n1000v(config-port-prof)# description "Port Profile for iSCSI multipath" n1000v(config-port-prof)#	Adds a description to the port profile. This description is automatically pushed to the vCenter Server. *profile description*: up to 80 ASCII characters **Note** If the description includes spaces, it must be surrounded by quotations.
Step 4	**vmware port-group [***name***]** **Example:** n1000v(config-port-prof)# vmware port-group VMK-port-profile n1000v(config-port-prof)#	Designates the port-profile as a VMware port group. The port profile is mapped to a VMware port group of the same name. When a vCenter Server connection is established, the port group created in Cisco Nexus 1000V is then distributed to the virtual switch on the vCenter Server. *name*: The VMware port group name.If you want to map the port profile to a different port group name, use the alternate name.
Step 5	**switchport mode access**] **Example:** n1000v(config-port-prof)# switchport mode access n1000v(config-port-prof)#	Designates that the interfaces are switch access ports (the default).
Step 6	**switchport access vlan** *vlanID* **Example:** n1000v(config-port-prof)# switchport access vlan 254 n1000v(config-port-prof)#	Assigns the system VLAN ID to the access port for this port profile. **Note** The VLAN assigned to this iSCSI port profile must be a system VLAN.
Step 7	**no shutdown** **Example:** n1000v(config-port-prof)# no shutdown n1000v(config-port-prof)#	Administratively enables all ports in the profile.

	Command	Purpose

Step 8	**system vlan** *vlanID*
	
	Example:
	`n1000v(config-port-prof)# system vlan`
	`254`
	`n1000v(config-port-prof)#`
		Adds the system VLAN to this port profile.
		
		This ensures that, when the host is added for the first time or rebooted later, the VEM will be able to reach the VSM. One of the uplink ports must have this VLAN in its system VLAN range.

Step 9	**capability iscsi-multipath**
	
	Example:
	`n1000v(config-port-prof)# capability`
	`iscsi-multipath`
	`n1000v(config-port-prof)#`
		Allows the port to be used for iSCSI multipathing.
		
		In vCenter Server, the iSCSI Multipath port profile must be selected and assigned to the VM kernel NIC port.

Step 10	**state enabled**
	
	Example:
	`n1000v(config-port-prof)# state enabled`
	`n1000v(config-port-prof)#`
		Enables the port profile.
		
		The configuration for this port profile is applied to the assigned ports, and the port group is created in the VMware vSwitch on the vCenter Server.

Step 11	**show port-profile name** *name*
	
	Example:
	`n1000v(config-port-prof)# show`
	`port-profile name multipath-profile`
	`n1000v(config-port-prof)#`
		(Optional) Displays the current configuration for the port profile.

Step 12	**copy running-config startup-config**
	
	Example:
	`n1000v(config-port-prof)# copy`
	`running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Creating VMKernel Ports and Attaching the Port Profile

You can use this procedure to create VMkernel ports and attach a port profile to them.

BEFORE YOU BEGIN

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

- You have already created a port profile by following the "Creating a Port Profile for VMkernel Ports" procedure on page 13-5.
- The VMKernel ports are created directly on the vSphere client.
- You cannot have two different port profiles for iSCSI traffic. All VMware VMkernel NICs for iSCSI must be attached to the same port profile.
- Create one VMkernel NIC for each physical NIC that carries the iSCSI VLAN. The number of paths to the storage device is the same as the number of VMkernel NICs created. The number of VMKs must match the number of physical uplink ports that are attached to the Cisco Nexus 1000V. Then, the Cisco Nexus 1000V can assign a VMK and pin it to a physical uplink.
- The VMkernel NICs you create in this procedure may also carry other VLANs.

Step 1   Create one VMkernel NIC for each physical NIC that carries the iSCSI VLAN.

For example, if you want to configure two paths, create two physical NICs on the Cisco Nexus 1000V DVS to carry the iSCSI VLAN. Create two vmknics for 2 paths.

Step 2   Attach the port profile configured with **capability iscsi-multipath** to the VMkernel ports.

Related Documents

Related Topic	Document Title
VMware SAN Configuration	VMware SAN Configuration Guide
Port Profile Configuration	*Cisco Nexus 1000V Port Profile Configuration Guide, Release 4.0(4)SV1(3)*
Interface Configuration	*Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(3)*
Complete command syntax, command modes, command history, defaults, usage guidelines, and examples for all Cisco Nexus 1000V commands.	*Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(3)*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

Feature History for iSCSI Multipath

Table 13-2 lists the release history for the iSCSI Multipath feature.

Table 13-2   Feature History for iSCSI Multipath

Feature Name	Releases	Feature Information
iSCSI Multipath	4.0(4)SV1(2)	Th iSCSI Multipath feature was added.

Configuration Limits

Use the following configuration limits with Cisco Nexus 1000V:

Component	Supported Limit	
Maximum Modules	66	
Virtual Ethernet (VEM)	64	
Virtual Supervisor (VSM)	2	
Hosts	64	
Active VLANs across all VEMs	512	
MACs over VLAN within a VEM	1024 (1K)	
vEthernet interfaces per port profile	1024 (1K)	
PVLAN	512	
vCenter Server connections	1 per VSM [1]	
	Per DVS	**Per Host**
Virtual Service Domains (VSD)	64	6
VSD Interfaces	2048	214
vEthernet interfaces	2K	216
Port Profiles	256	—
System Port Profiles	—	16
Port Channel	256	8
Physical Trunks	512	—
Physical NICs	—	32
vEthernet Trunks	256	8
ACL	128	16 [2]
ACEs per ACL	128	16 [2]
ACL Interfaces	2048	256
NetFlow Policies	32	8
NetFlow Interfaces	256	32
SPAN/ERSPAN Sessions	64	4

Component	Supported Limit	
QoS Policy-Map	128	16
QoS Class-Map	1024	128
QoS Interfaces	2048	256
Port Security	1600	216
MultiCast Groups	512	64

1. Only one connection to vCenter server is permitted at a time.

2. This number can be exceeded if VEM has available memory.

INDEX

U

users

    displaying   **7-1**

    sending messages   **7-2**

V

vCenter Server

    connect to   **4-2**

    disconnect from   **4-4**

    remove Nexus 1000V from   **4-4**

VLAN interfaces

    communicating between VLANs   **3-12, 3-14**

VLANs

    SVIs   **3-12, 3-14**

vmware port-group command   **3-11**

volatile:

    switch reboots   **6-3**