C H A P T E R 1

CLI Overview

This chapter prepares you to configure switches from the CLI (command-line interface). It also lists the information you need to have before you begin, and it describes the CLI command modes.

This chapter includes the following sections:

About the Switch Prompt

If you are connected to the console port when the switch boots up, you see the output shown in Example 1-1.

Note Refer to the *Cisco MDS 9100 Series Hardware Installation Guide*, the *Cisco MDS 9200 Series Hardware Installation Guide, the Cisco MDS 9216 Hardware Installation Guide*, or the *Cisco MDS 9500 Series Hardware Installation Guide* for installation and connection instructions.

Once the switch is powered on successfully, you see the default switch prompt (`switch#`). You can perform embedded CLI operations, access command history, and use command parsing functions at this prompt. The switch gathers the command string upon detecting an **Enter** (CR) and accepts commands from a terminal.

Example 1-1   Displays the Output When a Switch Boots Up

```
Auto booting bootflash:/boot-279 bootflash:/system_image;...
Booting kickstart image:bootflash:/boot-279....
..........................................Image verification OK

Starting kernel...
INIT: version 2.78 booting
Checking all filesystems..... done.
Loading system software
Uncompressing system image: bootflash:/system_image
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
INIT: Entering runlevel: 3

<<<<<<SAN OS bootup log messages>>>>>>

        ---- Basic System Configuration Dialog ----

This setup utility will guide you through the basic configuration of
the system. Use ctrl-c to abort configuration dialog at any prompt.

Basic management setup configures only enough connectivity for
management of the system.

Would you like to enter the basic configuration dialog (yes/no): yes

<<<<<<after configuration>>>>>>

switch login:
```

About the CLI Command Modes

Switches in the Cisco MDS 9000 Family have two main command modes—user EXEC mode and configuration mode. The commands available to you depend on the mode you are in. To obtain a list of available commands in either mode, type a question mark (?) at the system prompt.

Table 1-1 lists and describes the two commonly used modes, how to enter the modes, and the resulting system prompts. The system prompt helps you identify which mode you are in and hence, which commands are available to you.

Table 1-1   Frequently Used Switch Command Modes

| Mode	Description of Use	How to Access	Prompt
EXEC	Enables you to temporarily change terminal settings, perform basic tests, and display system information. **Note** Changes made in this mode are generally not saved across system resets.	At the switch prompt, enter the required EXEC mode command.	`switch#`
Configuration mode	Enables you to configure features that affect the system as a whole. **Note** Changes made in this mode are saved across system resets if you save your configuration. Refer to the Cisco *NX-OS 9000 Family Fundementals Configuration Guide* for further information.	From EXEC mode, enter the **config terminal** command.	`switch(config)#`

You can abbreviate commands and keywords by entering just enough characters to make the command unique from other commands. For example, you can abbreviate the **config terminal** command to **conf t**.

Understanding CLI Command Hierarchy

The CLI commands are organized hierarchically, with commands that perform similar functions grouped under the same level. For example, all commands that display information about the system, configuration, or hardware are grouped under the **show** command, and all commands that allow you to configure the switch are grouped under the **config terminal** command. Figure 1-1 illustrates a portion of the **config terminal** command hierarchy.

Figure 1-1   CLI Command Hierarchy Example
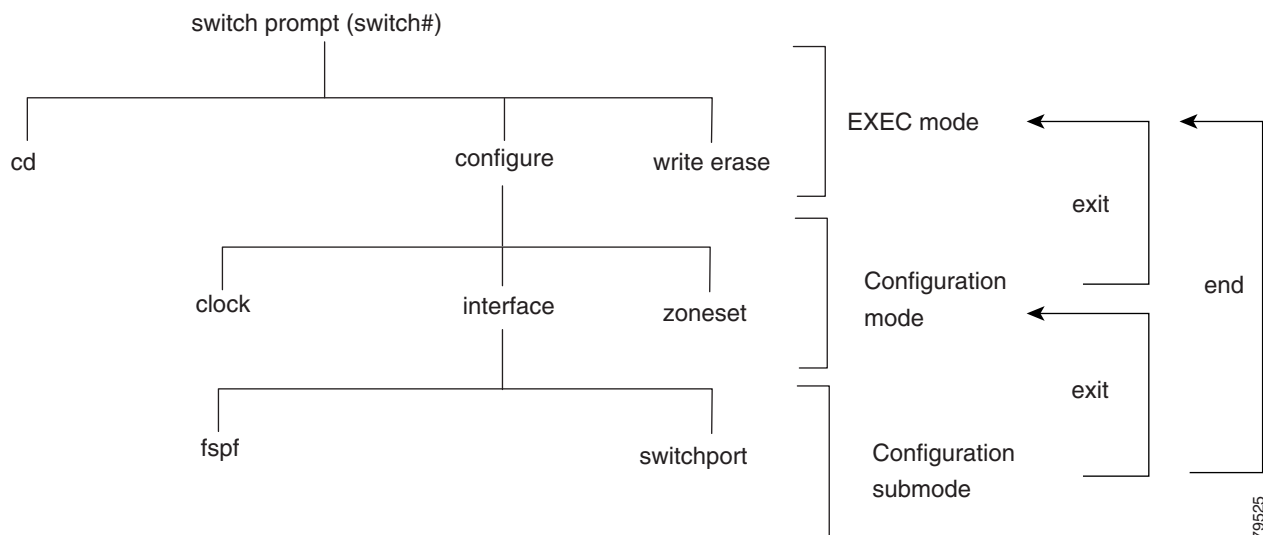
To execute a command, you enter the command by starting at the top level of the hierarchy. For example, to configure a Fibre Channel interface, use the **config terminal** command. Once you are in configuration mode, issue the **interface** command. When you are in the interface submode, you can query the available commands there.

The following example shows how to query the available commands in the interface submode:

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# interface fc1/1
switch(config-if)# ?
Interface configuration commands:
  channel-group  Add to/remove from a port-channel
  exit           Exit from this submode
  fcdomain       Enter the interface submode
  fspf           To configure FSPF related parameters
  no             Negate a command or set its defaults
  shutdown       Enable/disable an interface
  switchport     Configure switchport parameters
```

EXEC Mode Options

When you start a session on the switch, you begin in EXEC mode. Based on the role or group to which you belong, you have access to limited commands or to all commands (see the "Role-Based CLI" section on page 1-38). From the EXEC mode, you can enter configuration mode. Most of the EXEC commands are one-time commands, such as **show** commands, which display the current configuration status.

The next two sections list the EXEC mode commands for the Cisco MDS 9000 Family, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter. Not all EXEC mode commands that are supported on the Cisco MDS 9000 Family switches are available on the Cisco Fabric Switch for HP c-Class BladeSystem and the Cisco Fabric Switch for IBM BladeCenter.

EXEC Mode Commands for the Cisco MDS 9000 Family

```
switch# ?
Exec commands:
  attach         Connect to a specific linecard
  callhome       Callhome commands
  cd             Change current directory
  clear          Reset functions
  cli            CLI commands
  clock          Manage the system clock
  config         Enter configuration mode
  copy           Copy from one file to another
  debug          Debugging functions
  delete         Delete a file
  dir            List files in a directory
  discover       Discover information
  exit           Exit from the EXEC
  fcping         Ping an N-Port
  fctrace        Trace the route for an N-Port.
  find           Find a file below the current directory
  format         Format disks
  gunzip         Uncompresses LZ77 coded files
  gzip           Compresses file using LZ77 coding
  install        Upgrade software
  license        Enter the license configuration mode
  mkdir          Create new directory
  modem          Modem commands
  move           Move files
  no             Disable debugging functions
  ntp            Execute NTP commands
  out-of-service Make the current module out-of-service
  ping           Send echo messages
  port-channel   Port-Channel related commands
  purge          Deletes unused data
  pwd            View current directory
  reload         Reboot the entire box
  rmdir          Delete a directory
  run-script     Run shell scripts
  sdv            SDV test commands
  send           Send message to open sessions
  setup          Run the basic SETUP command facility
  show           Show running system information
  sleep          Sleep for the specified number of seconds
  ssh            SSH to another system
  system         System management commands
  tac-pac        Save tac information to a specific location
  tail           Display the last part of a file
  telnet         Telnet to another system
```

```
terminal        Set terminal line parameters
test            Test command
traceroute      Trace route to destination
undebug         Disable Debugging functions (See also debug)
update          Update license
write           Write current configuration
zone            Execute Zone Server commands
zoneset         Execute zoneset commands
```

EXEC Mode Commands for the Cisco Fabric Switch for HP c-Class BladeSystem and the Cisco Fabric Switch for IBM BladeCenter

```
switch# ?
Exec commands:
  attach          Connect to a specific linecard
  callhome        Callhome commands
  cd              Change current directory
  clear           Reset functions
  cli             CLI commands
  clock           Manage the system clock
  config          Enter configuration mode
  copy            Copy from one file to another
  debug           Debugging functions
  delete          Delete a file
  dir             List files in a directory
  discover        Discover information
  exit            Exit from the EXEC
  fcping          Ping an N-Port
  fctrace         Trace the route for an N-Port.
  find            Find a file below the current directory
  format          Format disks
  gunzip          Uncompresses LZ77 coded files
  gzip            Compresses file using LZ77 coding
  install         Upgrade software
  license         Enter the license configuration mode
  mkdir           Create new directory
  modem           Modem commands
  move            Move files
  no              Disable debugging functions
  ntp             Execute NTP commands
  out-of-service  Make the current module out-of-service
  ping            Send echo messages
  port-channel    Port-Channel related commands
  purge           Deletes unused data
  pwd             View current directory
  reload          Reboot the entire box
  rmdir           Delete a directory
  run-script      Run shell scripts
  send            Send message to open sessions
  setup           Run the basic SETUP command facility
  show            Show running system information
  sleep           Sleep for the specified number of seconds
  ssh             SSH to another system
  system          System management commands
  tac-pac         Save tac information to a specific location
  tail            Display the last part of a file
  telnet          Telnet to another system
  terminal        Set terminal line parameters
  test            Test command
  traceroute      Trace route to destination
  undebug         Disable Debugging functions (See also debug)
  update          Update license
```

```
write          Write current configuration
zone           Execute Zone Server commands
zoneset        Execute zoneset commands
```

Configuration Mode Options

Configuration mode allows you to make changes to the existing configuration. When you save the configuration, these commands are preserved across switch reboots. Once you are in configuration mode, you can enter interface configuration mode, zone configuration mode, and a variety of protocol-specific modes. Configuration mode is the starting point for all configuration commands. When you are in configuration mode, the switch expects configuration commands from the user.

The following example shows output from the **config terminal** command:

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)#
```

Not all configuration mode commands that are available on the Cisco MDS 9000 Family are available on the Cisco Fabric Switch for HP c-Class BladeSystem and the Cisco Fabric Switch for IBM BladeCenter.

The following two sections list the configuration mode commands for the Cisco MDS 9000 Family, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter. Not all configuration mode commands that are supported on the Cisco MDS 9000 Family switches are available on the Cisco Fabric Switch for HP c-Class BladeSystem and the Cisco Fabric Switch for IBM BladeCenter.

Configuration Mode Commands for the Cisco MDS 9000 Family

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ?
Configure commands:
  aaa              Configure aaa functions
  arp              [no] remove an entry from the ARP cache
  banner           Configure banner message
  boot             Configure boot variables
  callhome         Enter the callhome configuration mode
  cdp              CDP Configuration parameters
  cfs              CFS configuration commands
  cimserver        Modify cimserver configuration
  cli              CLI configuration commands
  clock            Configure time-of-day clock
  cloud            Configure Cloud Discovery
  cloud-discovery  Configure Cloud Discovery
  crypto           Set crypto settings
  device-alias     Device-alias configuration commands
  do               EXEC command
  dpvm             Configure Dynamic Port Vsan Membership
  end              Exit from configure mode
  exit             Exit from configure mode
  fabric-binding   Fabric Binding configuration
  fc-tunnel        Configure fc-tunnel
  fcalias          Fcalias configuration commands
  fcanalyzer       Configure cisco fabric analyzer
  fcc              Configure FC Congestion Control
  fcdomain         Enter the fcdomain configuration mode
  fcdroplatency    Configure switch or network latency
```

```
fcflow              Configure fcflow
fcid-allocation     Add/remove company id(or OUIs) from auto area list
fcinterop           Interop commands
fcip                Enable/Disable FCIP
fcns                Name server configuration
fcroute             Configure FC routes
fcrxbbcredit        Enable extended rx b2b credit configuration
fcs                 Configure Fabric Config Server
fcsp                Config commands for FC-SP
fctimer             Configure fibre channel timers
fdmi                Config commands for FDMI
ficon               Configure ficon information
fspf                Configure fspf
hw-module           Enable/Disable OBFL information
in-order-guarantee  Set in-order delivery guarantee
interface           Select an interface to configure
ip                  Configure IP features
ips                 Various sibyte module related commands
ipv6                Configure IPv6 features
iscsi               Enable/Disable iSCSI
islb                ISCSI server load-balancing
isns                Configure iSNS
isns-server         ISNS server
ivr                 Config commands for IVR
kernel              Kernel options
line                Configure a terminal line
logging             Modify message logging facilities
mcast               Configure multicast
no                  Negate a command or set its defaults
npiv                Nx port Id Virtualization (NPIV) feature enable
ntp                 NTP Configuration
port-security       Configure Port Security
port-track          Configure Switch port track config
power               Configure power supply
poweroff            Poweroff a module in the switch
qos                 QoS Configuration commands
radius              Configure RADIUS configuration
radius-server       Configure RADIUS related parameters
rib                 Configure RIB parameters
rmon                Remote Monitoring
role                Configure roles
rscn                Config commands for RSCN
san-ext-tuner       Enable/Disable San Extension Tuner tool
scheduler           Config commands for scheduler
scsi-target         Scsi-target configuration
snmp-server         Configure snmp server
span                Enter SPAN configuration mode
ssh                 Configure SSH parameters
switchname          Configure system's network name
system              System config command
tacacs+             Enable tacacs+
telnet              Enable telnet
tlport              Configure TL Port information
trunk               Configure Switch wide trunk protocol
username            Configure user information.
vsan                Enter the vsan configuration mode
wwn                 Set secondary base MAC addr and range for additional WWNs
zone                Zone configuration commands
zone-attribute-group  Zone attribute group commands
zoneset             Zoneset configuration commands
```

Configuration Mode Commands for the Cisco Fabric Switch for HP c-Class BladeSystem and the Cisco Fabric Switch for IBM BladeCenter

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ?
Configure commands:
  aaa                  Configure aaa functions
  arp                  [no] remove an entry from the ARP cache
  banner               Configure banner message
  boot                 Configure boot variables
  callhome             Enter the callhome configuration mode
  cdp                  CDP Configuration parameters
  cfs                  CFS configuration commands
  cimserver            Modify cimserver configuration
  cli                  CLI configuration commands
  clock                Configure time-of-day clock
  device-alias         Device-alias configuration commands
  do                   EXEC command
  dpvm                 Configure Dynamic Port Vsan Membership
  end                  Exit from configure mode
  exit                 Exit from configure mode
  fabric-binding       Fabric Binding configuration
  fcalias              Fcalias configuration commands
  fcanalyzer           Configure cisco fabric analyzer
  fcdomain             Enter the fcdomain configuration mode
  fcdroplatency        Configure switch or network latency
  fcflow               Configure fcflow
  fcid-allocation      Add/remove company id(or OUIs) from auto area list
  fcinterop            Interop commands
  fcns                 Name server configuration
  fcroute              Configure FC routes
  fcrxbbcredit         Enable extended rx b2b credit configuration
  fcs                  Configure Fabric Config Server
  fcsp                 Config commands for FC-SP
  fctimer              Configure fibre channel timers
  fdmi                 Config commands for FDMI
  fips                 Enable/Disable FIPS mode
  fspf                 Configure fspf
  hw-module            Enable/Disable OBFL information
  in-order-guarantee   Set in-order delivery guarantee
  interface            Select an interface to configure
  ip                   Configure IP features
  ipv6                 Configure IPv6 features
  kernel               Kernel options
  line                 Configure a terminal line
  logging              Modify message logging facilities
  mcast                Configure multicast
  no                   Negate a command or set its defaults
  npiv                 Nx port Id Virtualization (NPIV) feature enable
  ntp                  NTP Configuration
  port-security        Configure Port Security
  port-track           Configure Switch port track config
  power                Configure power supply
  poweroff             Poweroff a module in the switch
  qos                  QoS Configuration commands
  radius               Configure RADIUS configuration
  radius-server        Configure RADIUS related parameters
  rate-mode            Configure rate mode oversubscription limit
  rib                  Configure RIB parameters
  rlir                 Config commands for RLIR
  rmon                 Remote Monitoring
  role                 Configure roles
```

```
rscn                 Config commands for RSCN
scheduler            Config commands for scheduler
scsi-target          Scsi-target configuration
sdv                  Config commands for SAN Device Virtualization
snmp-server          Configure snmp server
span                 Enter SPAN configuration mode
ssh                  Configure SSH parameters
switchname           Configure system's network name
system               System config command
tacacs+              Enable tacacs+
tacacs-server        Configure TACACS+ server related parameters
telnet               Enable telnet
trunk                Configure Switch wide trunk protocol
username             Configure user information.
vsan                 Enter the vsan configuration mode
wwn                  Set secondary base MAC addr and range for additional WWNs
zone                 Zone configuration commands
zone-attribute-group Zone attribute group commands
zoneset              Zoneset configuration commands
```

Configuration mode, also known as terminal configuration mode, has several submodes. Each of these submodes places you deeper in the prompt hierarchy. When you type **exit**, the switch backs out one level and returns you to the previous level. When you type **end**, the switch backs out to the user EXEC level. You can also type **Ctrl-Z** in configuration mode as an alternative to typing **end**.

Note When in configuration mode, you can alternatively enter:

- **Ctrl-Z** instead of the **end** command
- **Ctrl-G** instead of the **exit** command

You can execute an EXEC mode command from a configuration mode or submode prompt. You can issue this command from any submode within the configuration mode. When in configuration mode (or in any submode), enter the **do** command along with the required EXEC mode command. The entered command is executed at the EXEC level and the prompt resumes its current mode level.

```
switch(config)# do terminal session-timeout 0
switch(config)#
```

In this example, **terminal session-timeout** is an EXEC mode command—you are issuing an EXEC mode command using the configuration mode **do** command.

The **do** command applies to all EXEC mode commands other than the **end** and **exit** commands. You can also use the help (?) and command completion (tab) features for EXEC commands when issuing a **do** command along with the EXEC command.

Table 1-2 lists some useful command keys that can be used in both EXEC and configuration modes:

Table 1-2   Useful Command Key Description

Command	Description
Ctrl-P	Up history
Ctrl-N	Down history
Ctrl-X-H	List history

Table 1-2   Useful Command Key Description (continued)

Command	Description
Alt-P	History search backwards
	Note The difference between **Tab** completion and **Alt- P** or **Alt-N** is that TAB completes the current word while **Alt- P** and **Alt-N** completes a previously-entered command.
Alt-N	History search forwards
Ctrl-G	Exit
Ctrl-Z	End
Ctrl-L	Clear screen

Table 1-3 displays the commonly used configuration submodes for the Cisco MDS 9000 Family.

Table 1-3   Submodes Within the Configuration Mode for the Cisco MDS 9000 Family

Submode Name	From Configuration Mode Enter	Submode Prompt	Configured Information
Call Home	**callhome**	`switch(config-callhome)#`	Contact, destination, and e-mail
FCS Registration	fcs register	`switch(config-fcs-register)#`	FCS attribute registration
	From FCS registration submode: platform name *name* **vsan** *vsan-id*	`switch(config-fcs-register-attrib)#`	Platform name and VSAN ID association
Fibre Channel alias	**fcalias name** *name* **vsan** *vsan-id*	`switch(config-fcalias)#`	Alias member
FSPF	**fspf config vsan** *vsan-id*	`switch(config-(fspf-config))#`	Static SPF computation, hold time, and autonomous region
Interface configuration	**interface** *type slot/port*	`switch(config-if)#`	Channel groups, Fibre Channel domains, FSPF parameters, switch port trunk and beacon information, and IP address
	From the VSAN or mgmt0 (management) interface configuration submode: **vrrp** *number*	`switch(config-if-vrrp)#`	Virtual router
iSCSI target	**iscsi virtual-target name**	`switch(config-iscsi-tgt)`	iSCSI virtual target
iSLB initiator	**islb initiator**	`switch(config-islb-init)#`	iSCSI server load balancing (iSLB) initiator
iSLB target	**islb virtual-target name**	`switch(config-islb-tgt)`	iSCSI server load balancing (iSLB) virtual target
Line console	**line console**	`switch(config-console)#`	Primary terminal console
VTY	line vty	`switch(config-line)#`	Virtual terminal line
Role	**role name**	`switch(config-role)#`	Rule
SPAN	**span session** *number*	`switch(config-span)#`	SPAN source, destination, and suspend session information
VSAN database	**vsan database**	`switch(config-vsan-db)#`	VSAN database

Cisco MDS 9000 Family Command Reference

Table 1-3   Submodes Within the Configuration Mode for the Cisco MDS 9000 Family (continued)

Submode Name	From Configuration Mode Enter	Submode Prompt	Configured Information
Zone	**zone name** *string* **vsan** *vsan-id*	`switch(config-zone)#`	Zone member
Zone set	**zoneset name** *name* **vsan** *vsan-id*	`switch(config-zoneset)#`	Zone set member
SDV virtual device	**sdv virtual-device name** *device-name* vsan *vsan-id*	`switch(config-sdv-virt-dev)#`	SAN Device Virtualization information

Table 1-4 displays the commonly used configuration submodes for the Cisco Fabric Switch for HP c-Class BladeSystem and the Cisco Fabric Switch for IBM BladeCenter.

Table 1-4   Submodes Within the Configuration Mode for the Cisco Cisco Fabric Switch for HP c-Class BladeSystem and the Cisco Fabric Switch for IBM BladeCenter

Submode Name	From Configuration Mode Enter	Submode Prompt	Configured Information
Call Home	**callhome**	`switch(config-callhome)#`	Contact, destination, and e-mail
FCS Registration	fcs register	`switch(config-fcs-register)#`	FCS attribute registration
	From FCS registration submode: platform name *name* **vsan** *vsan-id*	`switch(config-fcs-register-attrib)#`	Platform name and VSAN ID association
Fibre Channel alias	**fcalias name** *name* **vsan** *vsan-id*	`switch(config-fcalias)#`	Alias member
FSPF	**fspf config vsan** *vsan-id*	`switch(config-(fspf-config))#`	Static SPF computation, hold time, and autonomous region
Interface configuration	**interface** *type slot/port*	`switch(config-if)#`	Channel groups, Fibre Channel domains, FSPF parameters, switch port trunk and beacon information, and IP address
	From the VSAN or mgmt0 (management) interface configuration submode: **vrrp** *number*	`switch(config-if-vrrp)#`	Virtual router
Line console	**line console**	`switch(config-console)#`	Primary terminal console
VTY	line vty	`switch(config-line)#`	Virtual terminal line
Role	**role name**	`switch(config-role)#`	Rule
SPAN	**span session** *number*	`switch(config-span)#`	SPAN source, destination, and suspend session information
VSAN database	**vsan database**	`switch(config-vsan-db)#`	VSAN database
Zone	**zone name** *string* **vsan** *vsan-id*	`switch(config-zone)#`	Zone member
Zone set	**zoneset name** *name* **vsan** *vsan-id*	`switch(config-zoneset)#`	Zone set member
SDV virtual device	**sdv virtual-device name** *device-name* vsan *vsan-id*	`switch(config-sdv-virt-dev)#`	SAN device virtualization information

Note SPAN is only supported on external ports.

Navigating Through CLI Commands

To redisplay a command you previously entered, press the **Up Arrow** key. You can continue to press the **Up Arrow** key to see more previously issued commands. Similarly, you can press the **Down Arrow**, **Right Arrow**, **Left Arrow**, and **Delete** keys to navigate through the command history and to modify an existing command string.

Getting Help

In any command mode, you can get a list of available commands by entering a question mark (?).

```
switch# ?
```

To obtain a list of commands that begin with a particular character sequence, type in those characters followed immediately by the question mark (?). Do not include a space.

```
switch# co?
configure copy
```

To list keywords or arguments, enter a question mark in place of a keyword or argument. Include a space before the question mark. This form of help is called command syntax help, because it reminds you which keywords or arguments are applicable based on the commands, keywords, and arguments you have already entered.

```
switch# config ?
 terminal Configure the system from the terminal
```

Tip If you are having trouble entering a command, check the system prompt and enter the question mark (?) for a list of available commands. You might be in the wrong command mode or using incorrect syntax.

Command Completion

In any command mode, you can begin a particular command sequence and immediately press the **Tab** key to complete the rest of the command.

```
switch (config)# ro<Tab>
switch (config)# role <Tab>
switch (config)# role name
```

This form of help is called command completion, because it completes a word for you. If several options are available for the typed letters, all options that match those letters are presented:

```
switch(config)# fc<Tab>
fcalias        fcdomain        fcs
fcanalyzer     fcdroplatency   fcns          fctimer
fcc            fcinterop       fcroute
switch(config)# fcd<Tab>
fcdomain       fcdroplatency
switch(config)# fcdo<Tab>
switch(config)# fcdomain
```

Using the no and Default Forms of Commands

You can issue the **no** form of any command to perform the following actions:

- Undo a wrongly issued command.

    If you issue the **zone member** command, you can undo the results:

    ```
    switch(config)# zone name test vsan 1
    switch(config-zone)# member pwwn 12:12:12:12:12:12:12:12
    switch(config-zone)# no member pwwn 12:12:12:12:12:12:12:12
    WARNING: Zone is empty. Deleting zone test. Exit the submode.
    switch(config-zone)#
    ```

- Delete a created facility

    If you want to delete a zone that you created:

    ```
    switch(config)# zone name test vsan 1
    switch(config-zone)# exit
    switch(config)# no zone name test vsan 1
    switch(config)#
    ```

    You cannot delete a zone facility called test while residing in it. You must first exit the zone submode and return to configuration mode.

Port Names and Port Mapping

The Cisco Fabric Switch for HP c-Class BladeSystem has a different port naming convention from the Cisco MDS 9000 Family. It has eight external ports and they are labeled ext1 through ext8. It has 16 internal ports and they are labeled bay1 through bay16.

Table 1-5 shows the port mapping between the Cisco MDS 9000 Family and the Cisco Fabric Switch for HP c-Class BladeSystem.

Table 1-5   Port Mapping Between the Cisco MDS 9000 Family and the Cisco Fabric Switch for HP c-Class BladeSystem

Cisco MDS 9000 Family Port	Cisco Fabric Switch for HP c-Class BladeSystem Port	Port Group
fc1/1	ext8	PortGroup 1
fc1/2	bay6	
fc1/3	bay13	
fc1/4	bay5	
fc1/5	ext7	PortGroup 2
fc1/6	bay14	
fc1/7	bay15	
fc1/8	bay7	
fc1/9	bay4	PortGroup 3
fc1/10	ext1	
fc1/11	bay3	
fc1/12	bay11	
fc1/13	bay12	PortGroup 4

Table 1-5   *Port Mapping Between the Cisco MDS 9000 Family and the Cisco Fabric Switch for HP c-Class BladeSystem (continued)*

Cisco MDS 9000 Family Port	Cisco Fabric Switch for HP c-Class BladeSystem Port	Port Group
fc1/1	ext8	PortGroup 1
fc1/14	ext2	
fc1/15	bay2	
fc1/16	bay1	
fc1/17	bay10	PortGroup 5
fc1/18	ext3	
fc1/19	bay9	
fc1/20	ext 4	
fc1/21	bay16	PortGroup 6
fc1/22	bay8	
fc1/23	ext6	
fc1/24	ext5	

The Cisco Fabric Switch for IBM BladeCenter has a different port naming convention from the Cisco MDS 9000 Family. There are six external ports and they are labeled ext0 and ext15 through ext19. There are 14 internal ports and they are labeled bay1 through bay14. Table 1-6 shows the port mapping between the Cisco MDS 9000 Family switches and the Cisco Fabric Switch for IBM BladeCenter switches.

Table 1-6   *Port Mapping Between the Cisco MDS 9000 Family and the Cisco Fabric Switch for IBM BladeCenter*

Cisco MDS 9000 Family Port	Cisco Fabric Switch for IBM BladeCenter Port	Port Group
fc1/1	ext19	PortGroup 1
fc1/2	bay10	
fc1/3	bay11	
fc1/4	bay12	
fc1/5	ext18	PortGroup 2
fc1/6	bay9	
fc1/7	bay13	
fc1/8	bay14	
fc1/9	bay8	PortGroup 3
fc1/10	ext17	
fc1/11	bay6	
fc1/12	bay5	
fc1/13	bay7	PortGroup 4
fc1/14	ext16	

Table 1-6   Port Mapping Between the Cisco MDS 9000 Family and the Cisco Fabric Switch for IBM BladeCenter (continued)

Cisco MDS 9000 Family Port	Cisco Fabric Switch for IBM BladeCenter Port	Port Group
fc1/1	ext19	PortGroup 1
fc1/15	bay4	
fc1/16	bay2	
fc1/17	bay3	PortGroup 5
fc1/18	ext0	
fc1/19	bay1	
fc1/20	ext15	

When you enter commands that require port names for the Cisco Fabric Switch for HP c-Class BladeSystem or the Cisco Fabric Switch for IBM BladeCenter, use the appropriate naming convention from either Table 1-5 or Table 1-6. See Example 1-2 and Example 1-3 for typical commands that require port names.

Example 1-2   interface Command

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
chester-1(config)# interface ext2
chester-1(config-if)#
```

Example 1-3   show interface Command

```
switch# show interface bay 5
```

Entering CLI Commands

You can configure the software in one of two ways:

- You can create the configuration for the switch interactively by issuing commands at the CLI prompt.

- You can create an ASCII file containing a switch configuration and then load this file on the required system. You can then use the CLI to edit and activate the file.

Viewing Switch Configurations

You can view the ASCII form of the configuration file when required. To view the current configuration tree from the EXEC prompt, issue the **show running-config** command. If the running configuration is different from the startup configuration, issue the **show startup-config** command to view the ASCII version of the current startup configuration that was used to boot the switch.

You can also gather specific information on the entire switch configuration by issuing the relevant **show** commands. Configurations are displayed based a specified feature, interface, module, or VSAN. Available **show** commands for each feature are briefly described in this section and listed at the end of each chapter.

S e n d   d o c u m e n t a t i o n   c o m m e n t s   t o   m d s f e e d b a c k - d o c @ c i s c o . c o m

Examples 1-4 to 1-10 display a few **show** command examples.

Example 1-4   Displays Details on the Specified Interface

```
switch# show interface fc1/1
fc1/1 is up
    Hardware is Fibre Channel, 20:01:ac:16:5e:4a:00:00
    vsan is 1
    Port mode is E
    Speed is 1 Gbps
    Beacon is turned off
    FCID is 0x0b0100
        0 frames input, 0 bytes, 0 discards
        0 runts, 0 jabber, 0 too long, 0 too short
        0 input errors, 0 CRC, 0 invalid transmission words
        0 address id, 0 delimiter
        0 EOF abort, 0 fragmented, 0 unknown class
        0 frames output, 0 bytes, 0 discards
        Received 0 OLS, 0 LRR, 0 NOS, 0 loop inits
        Transmitted 0 OLS, 0 LRR, 0 NOS, 0 loop inits
```

Example 1-5   Displays the Software and Hardware Version

```
switch# show version
Cisco Storage Area Networking Operating System (SAN-OS) Software
TAC support: http://www.cisco.com/tac
Copyright (c) 2002-2006, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained herein are owned by
other third parties and are used and distributed under license.
Some parts of this software may be covered under the GNU Public
License or the GNU Lesser General Public License. A copy of
each such license is available at
http://www.gnu.org/licenses/gpl.html and
http://www.gnu.org/licenses/lgpl.html

Software
  BIOS:      version 1.1.0
  loader:    version 1.2(2)
  kickstart: version 3.0(3) [gdb]
  system:    version 3.0(3) [gdb]

  BIOS compile time:       10/24/03
  kickstart image file is: bootflash:///boot-3.0.3
  kickstart compile time:  9/15/2006 10:00:00 [10/02/2006 06:26:25]
  system image file is:    bootflash:///isan-3.0.3
  system compile time:     9/15/2006 10:00:00 [10/02/2006 06:45:25]

Hardware
  cisco MDS 9509 ("Supervisor/Fabric-1")
  Intel(R) Pentium(R) III CPU with 1028604 kB of memory.

  bootflash:  251904 kB
  slot0:      251904 kB

172.22.31.238   kernel uptime is 0 days 0 hour 2 minute(s) 2 second(s)

  Last reset at 744021 usecs after Tue Oct 21 14:55:11 1980
    Reason: Reset Requested by CLI command reload
    System version: 4.0(0.432)
    Service:
```

Example 1-6   Displays the Running Configuration

```
switch# show running
Building Configuration ...
 interface fc1/1
 interface fc1/2
 interface fc1/3
 interface fc1/4
 interface mgmt0
ip address 172.22.95.112 255.255.255.0
no shutdown
vsan database
boot system bootflash:system-237; sup-1
boot kickstart bootflash:boot-237 sup-1
callhome
ip default-gateway 172.22.95.1
switchname switch
trunk protocol enable
username admin password 5 /AFDAMD4B2xK2  role network-admin
```

Example 1-7   Displays the Difference between the Running and Startup Configuration

```
switch# show running diff
Building Configuration ...
*** Startup-config
--- Running-config
***************** 1,16 ****
  fcip enable
  ip default-gateway 172.22.91.1
  iscsi authentication none
  iscsi enable
! iscsi import target fc
  iscsi virtual-target name vt
    pWWN 21:00:00:04:cf:4c:52:c1
  all-initiator-permit
--- 1,20 ----
  fcip enable
+ aaa accounting logsize 500
+
+
+
  ip default-gateway 172.22.91.1
  iscsi authentication none
  iscsi enable
! iscsi initiator name junk
  iscsi virtual-target name vt
    pWWN 21:00:00:04:cf:4c:52:c1
  all-initiator-permit
```

Example 1-8   Displays the Configuration for a Specified Interface

```
switch# show running interface fc2/9
interface fc2/9
switchport mode E
no shutdown
```

Note   The **show running interface** command is different from the **show interface** command.

Example 1-9   Displays the Configuration for all Interfaces in a 16-Port Module

```
switch# show running interface fc2/10 - 12
interface fc2/10
switchport mode E
no shutdown

interface fc2/11
switchport mode E
no shutdown

interface fc2/12
switchport mode FL
no shutdown
```

Example 1-10   Displays the Configuration Per VSAN

```
switch# show running vsan 1
Building Configuration ...
zone name m vsan 1
  member pwwn 21:00:00:20:37:60:42:5c
  member pwwn 21:00:00:20:37:4b:00:a2
zoneset name m vsan 1
  member m
zoneset  activate name m vsan 1
```

Saving a Configuration

To save the configuration, enter the **copy running-config startup-config** command from the EXEC mode prompt to save the new configuration into nonvolatile storage. Once this command is issued, the running and the startup copies of the configuration are identical.

Clearing a Configuration

To clear a startup configuration, enter the **write erase** command from the EXEC mode prompt. Once this command is issued, the switch's startup configuration reverts to factory defaults. The running configuration is not affected. The **write erase** command erases the entire startup configuration with the exception of any configuration that affects the loader functionality.

The **write erase boot** command only erases the configuration that affects the loader functionality. The loader functionality configuration includes the boot variables and the mgmt0 IP configuration information (IP address, netmask and default gateway).

```
switch# write erase boot
This command will erase the boot variables and the ip configuration of interface mgmt 0
```

Searching and Filtering CLI Output

The Cisco MDS NX-OS CLI provides ways of searching through large amounts of command output and filtering output to exclude information you do not need. These features are enabled for **the show** command, which generally displays large amounts of data.

> **Note** The **show** command is always entered in EXEC mode.

When output continues beyond what is displayed on your screen, the Cisco MDS NX-OS CLI displays a --More-- prompt. Pressing **Return** displays the next line; pressing the **Spacebar** displays the next screen of output.

To search the **show** command output, use the following command in EXEC mode:

Command	Purpose
switch# **show** *any-command* **| begin** *pattern*	Begins unfiltered output of the **show** command with the first line that contains the pattern.

> **Note** Cisco MDS NX-OS documentation generally uses the vertical bar to indicate a choice of syntax. However, to search the output of the **show** command, you need to enter the pipe character (the vertical bar). In this section the pipe appears in bold (**\|**) to indicate that you should enter this character.

To filter **show** command output, use one of the following commands in EXEC mode:

Command	Purpose
switch# **show** *any-command* **| exclude** *pattern*	Displays output lines that do not contain the pattern.
switch# **show** *any-command* **| include** *pattern*	Displays output lines that contain the pattern.
switch# **show** *any-command* **| include** "*pattern1|pattern2*"	Displays output lines that contain either pattern1 or pattern2. **Note** The alternation patterns, "*pattern1|pattern2*", must appear within double quotes.
switch# **show** *any-command* **| include** *pattern [***next** *number] [***prev** *number]*	Displays output lines that contain the pattern. Optionally, using the **next** or **prev** parameter followed by a number also displays the designated number of lines.
switch# **show** *any-command* **| count** *number*	Displays the number lines of output in the display.

You can enter the **Ctrl-Z** key combination at any time to interrupt the output and return to EXEC mode. For example, you can enter the **show running-config | begin hostname** command to start the display of the running configuration file at the line containing the hostname setting, then use **Ctrl-Z** when you get to the end of the information you are interested in capturing. See the .

Multiple Filter Commands

Cisco MDS SAN-OS Release 2.1(1a) supports using multiple filters in the same **show** command output. This means you can use a combination of the available filters to format the output of any **show** command.

Note The maximum number of commands allowed is four. For example, you can enter a maximum of three filter commands or two filter commands and a redirection.

Cisco MDS SAN-OS Release 2.1(1a) also supports both filters and redirection in the same command. Now you can apply the required filters to the output of any command, and save the output using the file redirection. See the next section, "Searching and Filtering CLI Output Examples" section on page 1-21.

Searching and Filtering CLI Output Examples

The following is partial sample output of the **show running-config | begin** EXEC command. It begins displaying unfiltered output with the first line that contain the pattern `vsan`.

```
switch# show running-config | begin vsan
fcdomain fcid persistent vsan 1
fcdomain fcid persistent vsan 2
fcdomain fcid persistent vsan 3
fcdomain fcid persistent vsan 101
fcdomain fcid persistent vsan 102
fcdomain fcid database
  vsan 1 wwn 29:00:00:05:30:00:06:ea fcid 0x680000 dynamic
  vsan 1 wwn 28:0f:00:05:30:00:06:ea fcid 0x680001 dynamic
  vsan 1 wwn 28:10:00:05:30:00:06:ea fcid 0x680002 dynamic
  vsan 1 wwn 28:11:00:05:30:00:06:ea fcid 0x680003 dynamic
  vsan 1 wwn 28:12:00:05:30:00:06:ea fcid 0x680004 dynamic
  vsan 1 wwn 28:13:00:05:30:00:06:ea fcid 0x680005 dynamic
  vsan 1 wwn 28:14:00:05:30:00:06:ea fcid 0x680006 dynamic
  vsan 1 wwn 28:1f:00:05:30:00:06:ea fcid 0x680007 dynamic
  vsan 1 wwn 28:20:00:05:30:00:06:ea fcid 0x680008 dynamic
  vsan 1 wwn 21:00:00:e0:8b:05:76:28 fcid 0x680100 area dynamic
  vsan 1 wwn 20:c5:00:05:30:00:06:de fcid 0x680200 area dynamic
  vsan 1 wwn 28:2b:00:05:30:00:06:ea fcid 0x680012 dynamic
  vsan 1 wwn 28:2d:00:05:30:00:06:ea fcid 0x680013 dynamic
  vsan 1 wwn 28:2e:00:05:30:00:06:ea fcid 0x680014 dynamic
  vsan 1 wwn 28:2f:00:05:30:00:06:ea fcid 0x680015 dynamic
  vsan 1 wwn 28:30:00:05:30:00:06:ea fcid 0x680016 dynamic
--More--
```

The following is partial sample output of the **show tech-support** EXEC command. It begins displaying unfiltered output with the first line that contain the string `show interface brief`.

```
switch# show tech-support | begin "show interface brief"
------------------ show interface brief ------------------

-------------------------------------------------------------------------------
Interface  Vsan   Admin   Admin    Status        FCOT   Oper   Oper    Port
                  Mode    Trunk                          Mode   Speed   Channel
                          Mode                                  (Gbps)
-------------------------------------------------------------------------------
fc4/1      1      FX      --       sfpAbsent     --     --      --
fc4/2      1      FX      --       sfpAbsent     --     --      --
fc4/3      1      FX      --       sfpAbsent     --     --      --
fc4/4      1      FX      --       sfpAbsent     --     --      --
fc4/5      1      FX      --       up            swl    F       1       --
fc4/6      1      FX      --       sfpAbsent     --     --      --
fc4/7      1      FX      --       sfpAbsent     --     --      --
fc4/8      1      FX      --       sfpAbsent     --     --      --
fc4/9      1      E       on       notConnected  swl    --      --
fc4/10     1      FX      --       sfpAbsent     --     --      --
fc4/11     1      FX      --       sfpAbsent     --     --      --
```

```
fc4/12    1    FX    --    sfpAbsent    --    --    --
fc4/13    1    FX    --    sfpAbsent    --    --    --
fc4/14    1    FX    --    sfpAbsent    --    --    --
fc4/15    1    FX    --    sfpAbsent    --    --    --
--More--
```

The following is partial sample output of the **show running-config | exclude** EXEC command. It excludes any output line that contain the pattern `vsan`.

```
switch# show running-config | exclude vsan
version 2.1(1a)
poweroff module 9
fcdomain fcid database
ssm enable feature nasb interface fc4/1-4
ssm enable feature santap module 4
ssm enable feature nasb interface fc9/1-4
ssm enable feature santap interface fc9/5-8
ssm enable feature santap interface fc9/21-28
switchname switch
boot kickstart bootflash:/b2193 sup-1
boot system bootflash:/r2193 sup-1
boot kickstart bootflash:/b2193 sup-2
boot system bootflash:/r2193 sup-2
boot ssi bootflash:/laslc1.bin module 1
boot ssi bootflash:/laslc1.bin module 2
boot ssi bootflash:/laslc1.bin module 3
boot ssi bootflash:/laslc1.bin module 4
boot ssi bootflash:/laslc1.bin module 7
boot ssi bootflash:/laslc1.bin module 8
boot ssi bootflash:/laslc1.bin module 9
line console
  speed 38400
--More--
```

The following is partial sample output of the **show interface** EXEC command. It includes all output with the pattern `vsan`.

```
switch# show interface | include vsan
    Port vsan is 1
    Port vsan is 1
    Port vsan is 1
    Port vsan is 1
    Port vsan is 1
    Port vsan is 1
[information deleted]
```

The following is partial sample output of the **show interface** EXEC command. It includes all output with the pattern `FX` plus the next and previous five lines of output.

```
switch# show interface | include FX next 5 prev 5
fc4/1 is down (SFP not present)
    Hardware is Fibre Channel
    Port WWN is 20:c1:00:05:30:00:06:de
    Admin port mode is FX
    Port vsan is 1
    Receive data field Size is 2112
    Beacon is turned off
    5 minutes input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
    5 minutes output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
--
    0 transmit B2B credit remaining

fc4/2 is down (SFP not present)
    Hardware is Fibre Channel
```

```
        Port WWN is 20:c2:00:05:30:00:06:de
        Admin port mode is FX
        Port vsan is 1
        Receive data field Size is 2112
        Beacon is turned off
        5 minutes input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
        5 minutes output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
--
--More--
```

The following output of the **show running-config** EXEC command. It displays the number lines, or count, of the output.

```
switch# show running-config | count
    214
switch#
```

The following output of the **show interface brief** EXEC command. It displays the interfaces where the administration mode is FX.

```
switch# show interface brief | include FX
fc4/1      1    FX    --    sfpAbsent    --    --        --
fc4/2      1    FX    --    sfpAbsent    --    --        --
fc4/3      1    FX    --    sfpAbsent    --    --        --
fc4/4      1    FX    --    sfpAbsent    --    --        --
fc4/5      1    FX    --    up           swl  F     1    --
fc4/6      1    FX    --    sfpAbsent    --    --        --
fc4/7      1    FX    --    sfpAbsent    --    --        --
fc4/8      1    FX    --    sfpAbsent    --    --        --
fc4/10     1    FX    --    sfpAbsent    --    --        --
fc4/11     1    FX    --    sfpAbsent    --    --        --
fc4/12     1    FX    --    sfpAbsent    --    --        --
fc4/13     1    FX    --    sfpAbsent    --    --        --
fc4/14     1    FX    --    sfpAbsent    --    --        --
fc4/15     1    FX    --    sfpAbsent    --    --        --
fc4/16     1    FX    --    sfpAbsent    --    --        --
fc4/17     1    FX    --    sfpAbsent    --    --        --
fc4/18     1    FX    --    sfpAbsent    --    --        --
fc4/19     1    FX    --    sfpAbsent    --    --        --
fc4/20     1    FX    --    sfpAbsent    --    --        --
fc4/21     1    FX    --    sfpAbsent    --    --        --
fc4/22     1    FX    --    sfpAbsent    --    --        --
fc4/23     1    FX    --    sfpAbsent    --    --        --
fc4/24     1    FX    --    sfpAbsent    --    --        --
fc4/25     1    FX    --    sfpAbsent    --    --        --
fc4/26     1    FX    --    sfpAbsent    --    --        --
fc4/27     1    FX    --    sfpAbsent    --    --        --
fc4/28     1    FX    --    down         swl  --        --
fc4/29     1    FX    --    sfpAbsent    --    --        --
fc4/30     1    FX    --    sfpAbsent    --    --        --
fc4/31     1    FX    --    sfpAbsent    --    --        --
fc4/32     1    FX    --    sfpAbsent    --    --        --
switch#
```

The following output of the **show interface brief** EXEC command uses multiple filter commands. It display the number of interfaces, or count, where the administration mode is FX.

```
switch# show interface brief | include FX | count
    31
switch#
```

The following **show interface brief** EXEC command uses multiple filter commands to redirect the output where the administration mode is FX to the file named test.txt in the directory SavedData.

```
switch# show interface brief | include FX > SavedData\test.txt
switch# cd SavedData
switch# dir
      2263     Jan 12 18:53:41 2005  SavedData\test.txt

Usage for volatile://
      8192 bytes used
  20963328 bytes free
  20971520 bytes total
switch#
```

Displaying Users

The **show users** command displays all users currently accessing the switch.

```
switch# show users
admin    pts/7       Jan 12 20:56 (10.77.202.149)
admin    pts/9       Jan 12 23:29 (modena.cisco.com)
admin    pts/11      Jan 13 01:53 (dhcp-171-71-49-49.cisco.com)
```

Sending Messages to Users

The **send** command sends a message to all active CLI users currently using the switch. This message is restricted to 80 alphanumeric characters with spaces.

This example sends a warning message to all active users about the switch being shut down.

```
switch# send Shutting down the system in 2 minutes. Please log off.

Broadcast Message from admin@excal-112
        (/dev/pts/3) at 16:50 ...
Shutting down the system in 2 minutes. Please log off.
```

Using the ping Command

The **ping** command verifies the connectivity of a remote host or server by sending echo messages.

The syntax for this command is **ping** *<host or ip address>*

```
switch# ping 171.71.181.19
PING 171.71.181.19 (171.71.181.19): 56 data bytes
64 bytes from 171.71.181.19: icmp_seq=0 ttl=121 time=0.8 ms
64 bytes from 171.71.181.19: icmp_seq=1 ttl=121 time=0.8 ms

--- 171.71.181.19 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 0.8/0.8/0.8 ms
```

To abnormally terminate a ping session, type the **Ctrl-C** escape sequence.

Using traceroute

The **traceroute** command prints the routes taken by a specified host or IP address.

The syntax for this command is **traceroute** *<host or ip address>*

```
switch# traceroute www.cisco.com
traceroute to www.cisco.com (171.71.181.19), 30 hops max, 38 byte packets
 1  kingfisher1-92.cisco.com (172.22.92.2)  0.598 ms  0.470 ms  0.484 ms
 2  nubulab-gw1-bldg6.cisco.com (171.71.20.130)  0.698 ms  0.452 ms  0.481 ms
 3  172.24.109.185 (172.24.109.185)  0.478 ms  0.459 ms  0.484 ms
 4  sjc12-lab4-gw2.cisco.com (172.24.111.213)  0.529 ms  0.577 ms  0.480 ms
 5  sjc5-sbb4-gw1.cisco.com (171.71.241.174)  0.521 ms  0.495 ms  0.604 ms
 6  sjc12-dc2-gw2.cisco.com (171.71.241.230)  0.521 ms  0.614 ms  0.479 ms
 7  sjc12-dc2-cec-css1.cisco.com (171.71.181.5)  2.612 ms  2.093 ms  2.118 ms
 8  www.cisco.com (171.71.181.19)  2.496 ms * 2.135 ms
```

To abnormally terminate a traceroute session, enter **Ctrl-C**.

Setting the Switch's Shell Timeout

Use the **exec-timeout** command in configuration mode to configure the lifetime of all terminal sessions on that switch. When the time limit configured by this command is exceeded, the shell exits and closes that session. The syntax for this command from is **exec-timeout** *minutes*

The default is 30 minutes. You can configure different timeout values for a console or a virtual terminal line (VTY) session. You can set the **exec-timeout** value to 0 to disable this feature so the session remains active until you exit the switch. This change is saved in the configuration file.

- From the console:

```
switch(config)# line console
switch(config-console)# exec-timeout 60
```

Specifies the current console shell timeout to be 60 minutes.

- From a VTY session (Telnet or SSH):

```
switch(config)# line vty
switch(config-line)# exec-timeout 60
```

Specifies the current console shell timeout to be 60 minutes.

Displaying VTY Sessions

Use the **show line** command to display all configured VTY sessions:

```
switch# show line
line Console:
    Speed:       9600 bauds
    Databits:    8 bits per byte
    Stopbits:    1 bit(s)
    Parity:      none
    Modem In: Disable
    Modem Init-String -
        default : ATE0Q1&D2&C1S0=1\015
    Statistics: tx:5558511    rx:5033958     Register Bits:RTS|CTS|DTR|DSR|CD|RI
line Aux:
    Speed:       9600 bauds
    Databits:    8 bits per byte
    Stopbits:    1 bit(s)
    Parity:      none
    Modem In: Disable
    Modem Init-String -
        default : ATE0Q1&D2&C1S0=1\015
    Hardware Flowcontrol: ON
    Statistics: tx:35    rx:0    Register Bits:RTS|DTR
```

Send documentation comments to mdsfeedback-doc@cisco.com

Clearing VTY Sessions

Use the **clear line** command to close a specified VTY session:

```
switch# clear line Aux
```

Setting the Switch's Terminal Timeout

Use the **terminal session-timeout** command in EXEC mode to configure the automatic logout time for the current terminal session on that switch. When the time limit configured by this command is exceeded, the switch closes that session and exits.

The syntax for this command from is **terminal session-timeout** *minutes*

The default is 30 minutes. You can set the **terminal session-timeout** value to 0 to disable this feature so the terminal remains active until you choose to exit the switch. This change is not saved in the configuration file.

```
switch# terminal session-timeout 600
```

Specifies the terminal timeout to be 600 minutes for the current session.

Setting the Switch's Terminal Type

Use the **terminal terminal-type** command in EXEC mode to specify the terminal type for a switch:

The syntax for this command is **terminal terminal-type** *terminal-type*

```
switch# terminal terminal-type vt100
```

Specifies the terminal type. The *terminal-type* string is restricted to 80 characters and must be a valid type (for example vt100 or xterm). If a Telnet or SSH session specifies an unknown terminal type, the switch uses the vt100 terminal by default.

Setting the Switch's Terminal Length

To set the terminal screen length for the current session, use the **terminal length** command in EXEC mode. This command is specific to only the console port. Telnet and SSH sessions set the length automatically.

The syntax for this command is **terminal length** *lines*

```
switch# terminal length 20
```

Sets the screen length for the current session to 20 lines for the current terminal session. The default is 24 lines.

Setting the Switch's Terminal Width

To set the terminal screen width for the current session, use the **terminal width** command in EXEC mode. This command is specific to only the console port. Telnet and SSH sessions set the width automatically.

The syntax for this command is **terminal width** *columns*

```
switch# terminal width 86
```

Sets the screen length for the current session to 86 columns for the current terminal session. The default is 80 columns.

Displaying Terminal Settings

The show terminal command displays the terminal settings for the current session:

```
switch# show terminal
TTY: Type: "vt100"
Length: 24 lines, Width: 80 columns
Session Timeout: 525600 minutes
```

Using CLI Variables

The NX-OS CLI parser supports definition and use of variables in CLI commands. CLI variables can be used as follows:

- Entered directly on the command line.
- Passed to the child script initiated using the **run-script** command. The variables defined in the parent shell are available for use in the child **run-script** command process.
- Passed as command line arguments to the **run-script** command.

CLI variables have the following characteristics:

- You cannot reference a variables through another variable using nested references.
- You can define persistent variables that are available across switch reloads.
- You can reference only one predefined system variable, the TIMESTAMP variable.

User-Defined CLI Session Variables

You can define CLI variables the persist only for the duration of your CLI session using the **cli var name** command in EXEC mode. These CLI variables are useful for scripts that you execute periodically.

The following example shows how to create a user-defined CLI session variable:

```
switch# cli var name testinterface fc 1/1
```

You can reference a variable using the syntax **$(variable)**.

The following example shows how to reference a user-defined CLI session variable:

```
switch# show interface $(testinterface)
fc1/1 is up
    Hardware is Fibre Channel, SFP is short wave laser w/o OFC (SN)
    Port WWN is 20:01:00:0d:ec:0e:1d:00
    Admin port mode is auto, trunk mode is on
    snmp traps are enabled
    Port mode is F, FCID is 0x01000b
    Port vsan is 1
    Speed is 2 Gbps
    Transmit B2B Credit is 7
    Receive B2B Credit is 16
```

```
                    Receive data field Size is 2112
                    Beacon is turned off
                    5 minutes input rate 256 bits/sec, 32 bytes/sec, 1 frames/sec
                    5 minutes output rate 256 bits/sec, 32 bytes/sec, 1 frames/sec
                      232692 frames input, 7447280 bytes
                        0 discards, 0 errors
                        0 CRC,  0 unknown class
                        0 too long, 0 too short
                      232691 frames output, 7448692 bytes
                        0 discards, 0 errors
                      0 input OLS, 0 LRR, 0 NOS, 0 loop inits
                      1 output OLS, 1 LRR, 0 NOS, 1 loop inits
                      16 receive B2B credit remaining
                      7 transmit B2B credit remaining
```

Use the **show cli var** command to display user-defined CLI session variable:

The following example displays user-defined CLI session variables:

```
switch# show cli variables
VSH Variable List
----------------
TIMESTAMP="2005-10-24-21.29.33"
testinterface="fc 1/1"
```

Use the **cli no var name** command to remove user-defined CLI session variables:

The following example removes a user-defined CLI session variable:

```
switch# cli no var name testinterface
```

User-Defined CLI Persistent Variables

You can define CLI variables the persist across CLI sessions and switch reloads using the **cli var name** command in configuration mode. These CLI variables are configured in the configuration mode and are saved in the running configuration file.

The following example shows how to create a user-defined CLI persistent variable:

```
switch# config t
switch(config)# cli var name mgmtport mgmt 0
switch(config)# exit
switch#
```

You can reference a variable using the syntax **$(**variable**)**.

The following example shows how to reference a user-defined CLI persistent variable:

```
switch# show interface $(mgmtport)
mgmt0 is up
    Hardware is FastEthernet
    Address is 000e.38c6.2c6c
    Internet address is 10.10.10.1/24
    MTU 1500 bytes, BW 100 Mbps full Duplex
    288996 packets input, 97746406 bytes
      0 multicast frames, 0 compressed
      0 input errors, 0 frame, 0 overrun 0 fifo
    9089 packets output, 1234786 bytes, 0 underruns
      0 output errors, 0 collisions, 0 fifo
      0 carrier errors
```

Use the **show cli var** command to display user-defined CLI persistent variable.

The following example displays user-defined CLI persistent variables:

```
switch# show cli variables
VSH Variable List
----------------
TIMESTAMP="2005-10-24-21.37.13"
mgmtport="mgmt 0"
```

Use the **no cli var name** command in configuration mode to remove user-defined CLI persistent variables.

The following example removes a user-defined CLI persistent variable:

```
switch# config t
switch(config)# no cli var name mgmtport
```

System Defined Variables

Cisco MDS SAN-OS supports one predefined variable: TIMESTAMP. This variable refers to the time of execution of the command in the format YYYY-MM-DD-HH.MM.SS.

Note   The TIMESTAMP variable name is case sensitive. All letters must be uppercase.

The following example uses $(TIMESTAMP) when periodically gathering statistics into files using the command scheduler:

```
switch# config t l
switch(config)# scheduler enable
switch(config)# scheduler logfile size 16
switch(config)# scheduler job name j1
switch(config-job)# show interface mgmt0 | include mgmt > file
switch(config-job)# copy volatile:file bootflash:file.$(TIMESTAMP)
switch(config-job)# end
switch(config)#
```

The following example uses $(TIMESTAMP) when redirecting **show** command output to a file:

```
switch# show running-config > rcfg.$(TIMESTAMP)
Preparing to copy....done
switch# dir volatile:
       7231     Oct 03 20:20:42 2005  rcfg.2005-10-03-20.20.42

Usage for volatile://sup-local
8192 bytes used
20963328 bytes free
20971520 bytes total
```

Using Command Aliases

Command alias support has the following characteristics:

- Command aliases are global for all user sessions.
- Command aliases are persist across reboots.
- Commands being aliased must be typed in full without abbreviation.

- Command alias translation always takes precedence over any keyword in any configuration mode or submode.

- Command alias support is only available on the supervisor module, not the switching modules.

- Command alias configuration takes effect for other user sessions immediately.

- You cannot override the default command alias **alias,** which aliases the **show cli alias**.

- Nesting of command aliases is permitted to a maximum depth of 1. One command alias can refer to another command alias that must refer to a valid command, not to another command alias.

- A command alias always replaces the first command keyword on the command line.

- You can define command aliases for commands in any configuration submode or the EXEC mode.

Defining Command Aliases

You can define command aliases using the **cli alias name** command in configuration mode.

This following example shows how to define command aliases.

```
switch# config t
switch(config)# cli alias name gigint interface gigabitethernet
switch(config)# cli alias name shintbr show interface brief
switch(config)# cli alias name shfcintup "shintbr| include up | include fc"
```

You can display the command aliases defined on the switch using the **alias** default command alias.

The following example shows how to display the command aliases defined on the switch.

```
switch# alias
CLI alias commands
==================
alias      :show cli alias
gigint     :interface gigabitethernet
shintbr    :show interface brief
shfcintup  :shintbr | include up | include fc
```

About Flash Devices

Every switch in the Cisco MDS 9000 Family contains one internal bootflash (see Figure 1-2). The Cisco MDS 9500 Series additionally contains one external CompactFlash called slot0 (see Figure 1-2 and Figure 1-3).

Figure 1-2   Flash Devices in the Cisco MDS 9000 Supervisor Module
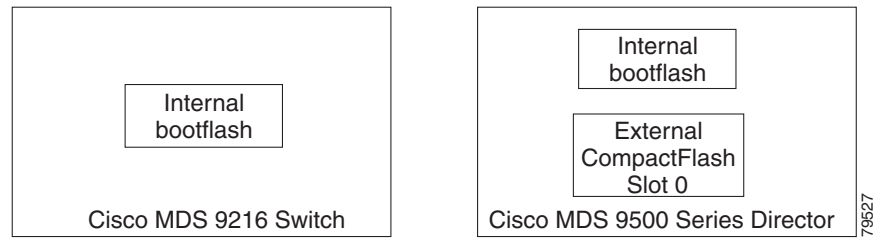
Figure 1-3   External CompactFlash in the Cisco MDS 9000 Supervisor Module
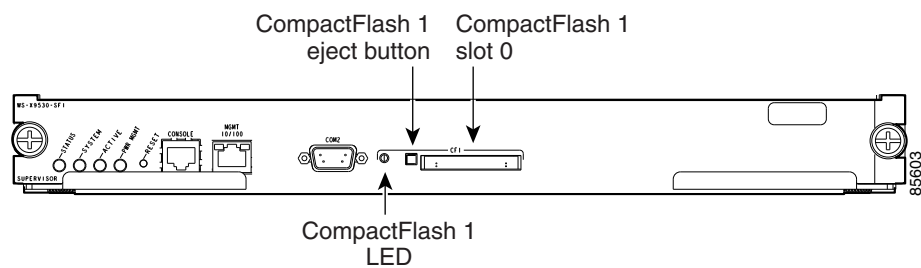
Internal bootflash:

All switches in the Cisco MDS 9000 Family have one internal bootflash: that resides in the supervisor or switching module.You have access to two directories within the internal bootflash: file system.

- The volatile: directory which provides temporary storage, and is also the default. Files in temporary storage (volatile:) are erased when the switch reboots.

- The bootflash (nonvolatile storage): directory which provides permanent storage. The files in bootflash are preserved through reboots and power outages.

External CompactFlash (Slot0)

Cisco MDS 9500 Series directors contain an additional external CompactFlash called slot0:

The external CompactFlash, an optional device for MDS 9500 Series directors, can be used for storing software images, logs, and core dumps.

Formatting Flash Disks and File Systems

By formatting a flash disk or a file system, you are essentially clearing out the contents of the disk or the file system and restoring it to its factory-shipped state (see the "About Flash Devices" section on page 1-30 and "Using the File System" section on page 1-32 for additional information).

Initializing bootflash:

When a switch is shipped, the **init system** command is already performed and you do not need to issue it again. Initializing the switch resets the entire internal disk and erases all data in the bootflash: partition. The internal disk is composed of several file systems with bootflash: being one of them. All files in bootflash: are erased and you must download the system and kickstart images again. After issuing an **init system** command, you don't have to format the bootflash: again since bootflash: is automatically formatted.

> **Note** The **init system** command also installs a new loader from the existing (running) kickstart image. You can access this command from the `switch(boot)#` prompt.

If bootflash: is found corrupted during a boot sequence, you will see the following message:

```
ERROR:bootflash: has unrecoverable error; please do "format bootflash:"
```

Use the **format bootflash:** command to only format the bootflash: file system. You can issue the **format bootflash:** command from either the `switch#` or the `switch(boot)#` prompts.

If you issue the **format bootflash:** command, you must download the kickstart and system images again.

Formatting Slot0:

Be sure to format an external CompactFlash device before using it to save files or images.

You can verify if the external CompactFlash device is formatted by inserting it into slot0: and issuing the **dir slot0:** command.

- If the external CompactFlash device is already formatted, you can see file system usage information (along with any existing files).

- If the external CompactFlash device is unformatted (corrupted), you will see the following message:

```
Device unavailable
```

In this case, you need to format the CompactFlash device using the **format slot0:** command.

> **Note** The slot0: file system cannot be accessed from the standby the `loader>` prompt or the `switch(boot)#` prompt, if the disk is inserted after booting the switch.

> **Caution** The Cisco MDS NX-OS software only supports Cisco-certified CompactFlash devices that are formatted using Cisco MDS switches. Using uncertified CompactFlash devices may result in unpredictable consequences; formatting CompactFlash devices using other platforms may result in errors.

Using the File System

The switch provides the following useful functions to help you manage software image files and configuration files:

Setting the Current Directory

The **cd** command changes the current directory level to a specified directory level. CLI defaults to the volatile: files system. This command expects a directory name input.

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

The syntax for this command is **cd** *directory name*

This example changes the current directory to the mystorage directory that resides in the slot0 directory:

```
switch# cd slot0:mystorage
```

This example changes the current directory to the mystorage directory that is in the current directory.

```
switch# cd mystorage
```

```
If the current directory is slot0:mydir, this command changes the current directory to
slot0:mydir/mystorage.
```

Displaying the Current Directory

The **pwd** command displays the current directory location. This example changes the directory and displays the current directory.

```
switch# cd bootflash:
switch# pwd
bootflash:
```

Listing the Files in a Directory

The **dir** command displays the contents of the current directory or the specified directory. The syntax for this command is **dir** *directory or file name*

This example shows how to list the files on the default volatile: file system:

```
switch# dir
          Usage for volatile: filesystem
                        0 bytes total used
                20971520 bytes free
                20971520 bytes available
```

Creating a New Directory

The **mkdir** command creates a directory at the current directory level or at a specified directory level.

The syntax for this command is **mkdir** *directory name*

This example creates a directory called test in the slot0 directory.

```
switch# mkdir slot0:test
```

This example creates a directory called test at the current directory level.

```
switch# mkdir test
```

```
If the current directory is slot0:mydir, this command creates a directory called
slot0:mydir/test.
```

Deleting an Existing Directory

The **rmdir** command deletes an existing directory at the current directory level or at a specified directory level. The directory must be empty to be deleted.

The syntax for this command is **rmdir** *directory name*

This example deletes the directory called test in the slot0 directory.

```
switch# rmdir slot0:test
```

This example deletes the directory called test at the current directory level.

```
switch# rmdir delete
```

```
If the current directory is slot0:mydir, this command deletes the slot0:mydir/test
directory.
```

Moving Files

The **move** command removes a file from the source directory and places it in the destination directory. If a file with the same name already exists in the destination directory, that file is overwritten by the moved file.

This example moves the file called samplefile from the slot0 directory to the mystorage directory.

```
switch# move slot0:samplefile slot0:mystorage/samplefile
```

```
This example moves a file from the current directory level.
```

```
switch# move samplefile mystorage/samplefile
```

```
If the current directory is slot0:mydir, this command moves slot0:mydir/samplefile to
slot0:mydir/mystorage/samplefile.
```

Copying Files

The **copy** command copies a file.

This example copies the file called samplefile from the external CompactFlash (slot0) directory to the mystorage directory.

```
switch# copy slot0:samplefile slot0:mystorage/samplefile
```

This example copies a file from the current directory level.

```
switch# copy samplefile mystorage/samplefile
```

If the current directory is slot0:mydir, this command copies slot0:mydir/samplefile to slot0:mydir/mystorage/samplefile.
You can also use the **copy** command to upload and download files from the slot0: or bootflash: file system to or from a FTP, TFTP, SFTP, or SCP server.

Deleting Files

The **delete** command deletes a specified file or the specified directory and all its contents.

This example shows how to delete a file from the bootflash: directory (assuming you are already in the bootflash: directory):

```
switch# delete dns_config.cfg
```

This example shows how to delete a file from an external CompactFlash (slot0):

```
switch# delete slot0:dns_config.cfg
```

This example deletes the entire `my-dir` directory and all its contents:

```
switch# delete bootflash:my-dir
```

⚠
Caution   If you specify a directory, the **delete** command deletes the entire directory and all its contents.

Displaying File Contents

The **show file** command displays the contents of a specified file in the file system.

The syntax for this command is **show file** *file_name*

This example displays the contents of the test file that resides in the slot0 directory:

```
switch# show file slot0:test
config t
Int fc1/1
no shut
end
show int
```

This example displays the contents of a file residing in the current directory.

```
switch# show file myfile
```

Saving Command Output to a File

You can force all screen output to go to a file by appending **>** *filename* to any command. For example, enter **show interface > samplefile** at the EXEC mode switch prompt to save the interface configuration to *samplefile*—a file created at the same directory level. At the EXEC mode switch prompt, issue a **dir** command to view all files in this directory, including the recently saved *samplefile*.

> **Note**   Redirection is allowed only if the current directory is on the `volatile:` (default) or `slot0:` file systems. Redirection is not allowed if the current directory is on the `bootflash:` file system. The current directory can be viewed using the **pwd** command and changed using the **cd** command.

Directing show Command Output to a File

You can direct **show** command output to a file, either on the volatile file system, on slot0 CompactFlash memory, or on a remote server.

The following example shows how to direct the **show running-config** output to a file on the volatile file system.

```
switch1# show running-config > volatile:switch1-run.cfg
```

The following example shows how to direct the **show running-config** output to a file on slot0 CompactFlash memory.

```
switch2# show running-config > slot0:switch2-run.cfg
```

The following example shows how to direct the **show running-config** output to a file on a TFTP server.

```
switch3# show running-config > tftp://10.10.1.1/home/suser/switch3-run.cfg
Preparing to copy...done
```

Compressing and Uncompressing Files

The **gzip** command compresses (zips) the specified file using LZ77 coding.

This example directs the output of the show tech-support command to a file (Samplefile) and then zips the file and displays the difference in the space used up in the volatile: directory:

```
switch# show tech-support > Samplefile
Building Configuration ...
switch# dir
    1525859     Jul 04 00:51:03 2003 Samplefile
Usage for volatile://
    1527808 bytes used
   19443712 bytes free
   20971520 bytes total
switch# gzip volatile:Samplefile
switch# dir
     266069     Jul 04 00:51:03 2003 Samplefile.gz
Usage for volatile://
     266240 bytes used
   20705280 bytes free
   20971520 bytes total
```

The **gunzip** command uncompresses (unzips) LZ77 coded files.

This example unzips the file that was compressed in the previous example:

```
switch# gunzip samplefile
/volatile/samplefile.gz: No such file or directory
switch# gunzip Samplefile
switch# dir
    1525859    Jul 04 00:51:03 2003 Samplefile
Usage for volatile://
    1527808 bytes used
   19443712 bytes free
   20971520 bytes total
```

Displaying the Last Line in a File

The **tail** command displays the last lines (tail end) of a specified file.

The syntax for this command is **tail** *<file name> [<number of lines>]*

```
switch# tail mylog 10
```

You see the last 10 lines of the mylog file.

Executing Commands Specified in a Script

The **run-script** command executes the commands specified in a file. To use this command, be sure to create the file and specify commands in the required order.

Note You cannot create the script files at the switch prompt. You can create the script file on an external machine and copy it the bootflash: directory. This section assumes that the script file resides in the bootflash: directory.

The syntax for this command is **run-script** *file_name.*

This example displays the CLI commands specified in the testfile that resides in the slot0 directory:

```
switch# show file slot0:testfile
conf t
interface fc 1/1
no shutdown
end
sh interface fc1/1
```

This file output is in response to the **run-script** command executing the contents in the testfile file:

```
switch# run-script slot0:testfile
'conf t'
Enter configuration commands, one per line. End with CNTL/Z.

'interface fc1/1'

'no shutdown'

'end'

'sh interface fc1/1'
fc1/1 is down (Fcot not present)
    Hardware is Fibre Channel
    Port WWN is 20:01:00:05:30:00:48:9e
```

```
Admin port mode is auto, trunk mode is on
vsan is 1
Beacon is turned off
Counter Values (current):
  0 frames input, 0 bytes, 0 discards
  0 runts, 0 jabber, 0 too long, 0 too short
  0 input errors, 0 CRC, 0 invalid transmission words
  0 address id, 0 delimiter
  0 EOF abort, 0 fragmented, 0 unknown class
  0 frames output, 0 bytes, 0 discards
  Received 0 OLS, 0 LRR, 0 NOS, 0 loop inits
  Transmitted 0 OLS, 0 LRR, 0 NOS, 0 loop inits
Counter Values (5 minute averages):
  0 frames input, 0 bytes, 0 discards
  0 runts, 0 jabber, 0 too long, 0 too short
  0 input errors, 0 CRC, 0 invalid transmission words
  0 address id, 0 delimiter
  0 EOF abort, 0 fragmented, 0 unknown class
  0 frames output, 0 bytes, 0 discards
  Received 0 OLS, 0 LRR, 0 NOS, 0 loop inits
  Transmitted 0 OLS, 0 LRR, 0 NOS, 0 loop inits
```

Setting the Delay Time

The **sleep** command delays an action by a specified number of seconds.

The syntax for this command is **sleep** *<seconds>*

```
switch# sleep 30
```

You will see the switch prompt return after 30 seconds.

This command is useful within scripts. For example, if you create a script called test-script:

```
switch# show file slot0:test-script
discover scsi-target remote
sleep 10
show scsi-target disk

switch# run-script slot0:test-script
```

When you execute the slot0:test-script, the switch software executes the **discover scsi-target remote** command, and then waits for 10 seconds before executing the **show scsi-target disk command**.

Role-Based CLI

By default, two roles exist in all switches:

- Network operator—Has permission to view the configuration.
- Network administrator—Has permission to execute all commands and to set up to 64 permission levels based on user roles and groups.

When you execute a command, perform command completion, or obtain context sensitive help, the switch software allows the operation to progress if you have the correct permission as specified in the description of the command.

Using Valid Formats and Ranges

> **Note** Do not enter ellipsis (...), vertical bar (|), less or great (< >), bracket ([ ]), or braces ({ }) in command lines. These characters have special meaning in Cisco MDS SAN-OS text strings.

Some commands require a MAC address, IP address, or IDs that must be designated in a standard format or given a range. See Table 1-7.

Table 1-7   Valid Formats and Ranges

Address	Description	Valid Format Example	Range
MAC address	6 bytes in hexadecimal format separated by colons (not case-sensitive) .	00:00:0c:24:d2:Fe	—
IP address	32 bytes, written as 4 octets separated by periods (dotted decimal format) that are made up of a network section, an optional netmask section, and a host section.	126.2.54.1	—
VSAN	Integer that specifies the VSAN.	7	1 to 4093
VLAN	Integer that specifies the VLAN.	11	1 to 4093
Port WWN (pWWN)	Eight hexadecimal numbers separated by colons (not case-sensitive).	12:34:56:78:9A:BC:dE:F1	—
Node WWN (nWWN)	Eight hexadecimal numbers separated by colons (not case-sensitive).	12:34:56:78:9A:BC:dE:F1	—
LUN	8 bytes in hexadecimal format separated by colons. A minimum of two hex characters are acceptable. The valid format is hhhh[:hhhh[:hhhh[:hhhh]]].	64 (100d = 64h)	—
FCID	Six character hexadecimal value prepended by 0x.	0xabc123	—
Domain ID	Integer that specifies the domain.	7	1 to 239
Timers	Integer that specifies timers in milliseconds for latency, FC time out values (TOV).	100	0 to 2147483647
Switching module	Slot in which the applicable switching module resides.	1	1 to 15
Switch priority	Integer specifying switch priority.	5	1 to 254
Channel group	Integer that specifies a PortChannel group addition.	1	1 to 100
Fabric Shortest Path First (FSPF)	Integer that specifies the hold time (in milliseconds) before making FSPF computations.	1000	0 to 65535
Fabric Analyzer	The allowed range for the frame size limit in bytes.	64	64 to 65536
Fabric Analyzer captures	An example of 10 frames, limits the number of frames captured to 10.	10	0 to 2147483647
FCIP profile	Integer that specifies the FCIP profile.	101	1 to 255
TCP retransmit time	Integer that specifies the minimum retransmit time for the TCP connection in milliseconds	300	250 to 5000

Table 1-7   Valid Formats and Ranges (continued)

Address	Description	Valid Format Example	Range
Keepalive timeout	Integer that specifies the TCP connection's keepalive timeout in seconds.	60	1 to 7200
TCP retransmissions	Integer that specifies the maximum number of TCP transmissions.	6	1 to 8
PMTU	Integer that specifies the path MTU reset time in seconds.	90	60 to 3600
TCP buffer size	Integer that specifies the advertised TCP buffer size in KB.	5000	0 to 8192
Traffic burst size	Integer that specifies the maximum burst size in KB.	30	10 to 100
Peer TCP port	Integer that specifies the TCP port number.	3000	0 to 65535
Acceptable time difference	Integer that specifies the acceptable time difference in milliseconds for a packet being accepted.	4000	1 to 60,000
iSCSI pWWN allocation	Integer that specifies the number of pWWNs that must be allocated to an iSCSI initiator.	2	1 to 64
CDP refresh and hold time	Integer that specifies the refresh time interval and the hold time in seconds for the CDP protocol.	60	5 to 255

Using Debug Commands

Caution   Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

All **debug** commands are entered in privileged EXEC mode, and most **debug** commands take no arguments. Use the **show debugging** command to display the state of each debugging option.

To list and see a brief description of all the debugging command options, enter the command **debug ?** at the command line in privileged EXEC mode. For example:

```
switch# debug ?
```

Not all debugging commands listed in the **debug ?** output are described in this document. Commands are included here based on their usefulness in assisting you to diagnose network problems. Commands not included are typically used internally by Cisco engineers during the development process and are not intended for use outside the Cisco environment.

To enable all system diagnostics, enter the **debug all** command at the command line in privileged EXEC mode. For example:

```
switch# debug all
```

To turn off all diagnostic output, enter the **no debug all** command at the command line in privileged EXEC mode. For example:

```
switch# no debug all
```

Using the **no debug all** command is a convenient way to ensure that you have not accidentally left any **debug** commands turned on.

⚠

Caution   Because debugging output takes priority over other network traffic, and because the **debug all** command generates more output than any other **debug** command, it can severely diminish the performance of the router or even render it unusable. In virtually all cases, it is best to use more specific **debug** commands.

Generating debug Command Output

Enabling a **debug** command can result in output similar to the following example for the **debug modem** command:

```
Router# debug modem

15:25:51: TTY4: DSR came up
15:25:51: tty4: Modem: IDLE->READY
15:25:51: TTY4: Autoselect started
15:27:51: TTY4: Autoselect failed
15:27:51: TTY4: Line reset
15:27:51: TTY4: Modem: READY->HANGUP
15:27:52: TTY4: dropping DTR, hanging up
15:27:52: tty4: Modem: HANGUP->IDLE
15:27:57: TTY4: restoring DTR
15:27:58: TTY4: DSR came up
```

The router continues to generate such output until you enter the corresponding **no debug** command (in this case, the **no debug modem** command).

If you enable a **debug** command and no output is displayed, consider the following possibilities:

- The router may not be properly configured to generate the type of traffic you want to monitor. Use the **more system:running-config** EXEC command to check its configuration.

- Even if the router is properly configured, it may not generate the type of traffic you want to monitor during the particular period that debugging is turned on. Depending on the protocol you are debugging, you can use commands such as the TCP/IP **ping** EXEC command to generate network traffic.

Redirecting debug and Error Message Output

By default, the network server sends the output from **debug** commands and system error messages to the console. If you use this default, monitor debug output using a virtual terminal connection, rather than the console port.

To redirect debug output, use the **logging** command options within configuration mode as described in the following sections.

Possible destinations include the console, virtual terminals, internal buffer, and UNIX hosts running a syslog server. The syslog format is compatible with 4.3 Berkeley Standard Distribution (BSD) UNIX and its derivatives.

Note Be aware that the debugging destination you use affects system overhead. Logging to the console produces very high overhead, whereas logging to a virtual terminal produces less overhead. Logging to a syslog server produces even less, and logging to an internal buffer produces the least overhead of any method.

To configure message logging, you need to be in configuration command mode. To enter this mode, use the **configure terminal** command at the EXEC prompt.

Enabling Message Logging

To enable message logging to all supported destinations other than the console, enter the following command:

logging on

The default condition is **logging on**.

To direct logging to the console only and disable logging output to other destinations, enter the following command:

no logging on

Setting the Message Logging Levels

You can set the logging levels when logging messages to the following devices:

- Console
- Monitor
- Syslog server

Table 1-8 lists and briefly describes the logging levels and corresponding keywords you can use to set the logging levels for these types of messages. The highest level of message is level 0, *emergencies*. The lowest level is level 7, *debugging*, which also displays the greatest amount of messages. For information about limiting these messages, see sections later in this chapter.

Table 1-8   Message Logging Keywords and Levels

Level	Keyword	Description	Syslog Definition
0	**emergencies**	System is unusable.	LOG_EMERG
1	**alerts**	Immediate action is needed.	LOG_ALERT
2	**critical**	Critical conditions exist.	LOG_CRIT
3	**errors**	Error conditions exist.	LOG_ERR
4	**warnings**	Warning conditions exist.	LOG_WARNING
5	**notification**	Normal, but significant, conditions exist.	LOG_NOTICE
6	**informational**	Informational messages.	LOG_INFO
7	**debugging**	Debugging messages.	LOG_DEBUG

Limiting the Types of Logging Messages Sent to the Console

To limit the types of messages that are logged to the console, use the **logging console** router configuration command. The full syntax of this command follows:

logging console *level*

no logging console

The **logging console** command limits the logging messages displayed on the console to messages up to and including the specified severity level, which is specified by the *level* argument. Keywords are listed in order from the most severe level to the least severe.

The **no logging console** command disables logging to the console.

The following example sets console logging of messages at the **debugging** level, which is the least severe level and which displays all logging messages:

```
logging console debugging
```

Logging Messages to an Internal Buffer

The default logging device is the console; all messages are displayed on the console unless otherwise specified.

To log messages to an internal buffer, use the **logging buffered** router configuration command. The full syntax of this command follows:

logging buffered

no logging buffered

The **logging buffered** command copies logging messages to an internal buffer instead of writing them to the console. The buffer is circular in nature, so newer messages overwrite older messages. To display the messages that are logged in the buffer, use the **show logging** privileged EXEC command. The first message displayed is the oldest message in the buffer.

The **no logging buffered** command cancels the use of the buffer and writes messages to the console (the default).

Limiting the Types of Logging Messages Sent to Another Monitor

To limit the level of messages logged to the terminal lines (monitors), use the **logging monitor** router configuration command. The full syntax of this command follows:

logging monitor *level*

no logging monitor

The **logging monitor** command limits the logging messages displayed on terminal lines other than the console line to messages with a level up to and including the specified *level* argument. To display logging messages on a terminal (virtual console), use the **terminal monitor** privileged EXEC command.

The **no logging monitor** command disables logging to terminal lines other than the console line.

The following example sets the level of messages displayed on monitors other than the console to **notification**:

```
logging monitor notification
```

Logging Messages to a UNIX Syslog Server

To log messages to a syslog server host, use the **logging host** global configuration command. The full syntax of this command follows:

logging host {*ip-address* | *host-name*} [**xml**]

no logging host {*ip-address* | *host-name*} [**xml**]

The **logging host** command identifies a syslog server host that is to receive logging messages. The *ip-address* argument is the IP address of the host. By issuing this command more than once, you build a list of syslog servers that receive logging messages.

The **no logging host** command deletes the syslog server with the specified address from the list of syslogs.

Limiting Messages to a Syslog Server

To limit the number of messages sent to syslog servers, use the **logging trap** router configuration command. The full syntax of this command follows:

logging trap *level*

no logging trap

The **logging trap** command limits the logging messages sent to syslog servers to logging messages with a level up to and including the specified *level* argument.

To send logging messages to a syslog server, specify its host address with the **logging host** command.

The default trap level is **informational**.

The **no logging trap** command returns the trap level to the default.

The current software generates the following categories of syslog messages:

- Error messages at the **emergencies** level.
- Error messages at the **alerts** level.
- Error messages at the **critical** level.
- Error messages about software or hardware malfunctions, displayed at the **errors** level.
- Interface up/down transitions and system restart messages, displayed at the **notification** level.
- Reload requests and low-process stack messages, displayed at the **informational** level.
- Output from the **debug** commands, displayed at the **debugging** level.

The **show logging** privileged EXEC command displays the addresses and levels associated with the current logging setup. The command output also includes ancillary statistics.

Example of Setting Up a UNIX Syslog Daemon

To set up the syslog daemon on a 4.3 BSD UNIX system, include a line such as the following in the file */etc/syslog.conf*:

```
local7.debugging /usr/adm/logs/tiplog
```

The **local7** keyword specifies the logging facility to be used.

The **debugging** keyword specifies the syslog level. See Table 1-8 for other keywords that can be listed.

The UNIX system sends messages at or above this level to the specified file, in this case */usr/adm/logs/tiplog*. The file must already exist, and the syslog daemon must have permission to write to it.

For the System V UNIX systems, the line should read as follows:

```
local7.debug /usr/admin/logs/cisco.log
```