show gsr through show monitor event trace

show gsr through show monitor event trace

show gsr

To display hardware information on the Cisco 12000 series Gigabit Switch Routers (GSRs), use the **show gsr** command in EXEC mode.

show gsr [**chassis-info** [**details**]]

Syntax Description

chassis-info	(Optional) Displays backplane NVRAM information.
details	(Optional) In addition to the information displayed, this option includes hexadecimal output of the backplane NVRAM information.

Command Modes

EXEC

Command History

Release	Modification
11.2GS	This command was introduced to support the Cisco 12000 series GSRs.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Use this command to determine the type of hardware installed in your Cisco 12000 series GSR router.

Examples

The following is sample output from the **show gsr** command for a Cisco 12012 router. This command shows the type and state of the card installed in the slot.

```
Router# show gsr
Slot 0  type  = Route Processor
        state = IOS Running  MASTER
Slot 7  type  = 1 Port Packet Over SONET OC-12c/STM-4c
        state = Card Powered
Slot 16 type  = Clock Scheduler Card
        state = Card Powered  PRIMARY CLOCK
```

The following is sample output from the **show gsr chassis-info** command for a Cisco 12012 router:

```
Router# show gsr chassis-info
Backplane NVRAM [version 0x20] Contents -
  Chassis: type 12012 Fab Ver: 1
    Chassis S/N: ZQ24CS3WT86MGVHL
  PCA: 800-3015-1  rev: A0  dev: 257  HW ver: 1.0
    Backplane S/N: A109EXPR75FUNYJK
  MAC Addr: base 0000.EAB2.34FF  block size: 1024
  RMA Number: 0x5F-0x2D-0x44  code: 0x01  hist: 0x1A
```

show gt64010 (7200)

To display all GT64010 internal registers and interrupt status on the Cisco 7200 series routers, use the **show gt64010** command in EXEC mode.

show gt64010

Syntax Description

This command has no arguments or keywords.

Command Modes

EXEC

Command History

Release	Modification
11.2	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

This command displays information about the CPU interface, DRAM/device address space, device parameters, direct memory access (DMA) channels, timers and counters, and protocol control information (PCI) internal registers. The information is generally useful for diagnostic tasks performed by technical support only.

Examples

The following is a partial sample output for the **show gt64010** command:

```
Router# show gt64010
GT64010 Channel 0 DMA:
 dma_list=0x6088C3EC, dma_ring=0x4B018480, dma_entries=256
 dma_free=0x6088CECC, dma_reqt=0x6088CECC, dma_done=0x6088CECC
 thread=0x6088CEAC, thread_end=0x6088CEAC
 backup_thread=0x0, backup_thread_end=0x0
 dma_working=0, dma_complete=6231, post_coalesce_frames=6231
 exhausted_dma_entries=0, post_coalesce_callback=6231
GT64010 Register Dump: Registers at 0xB4000000
CPU Interface:
 cpu_interface_conf   : 0x80030000 (b/s 0x00000380)
 addr_decode_err      : 0xFFFFFFFF (b/s 0xFFFFFFFF)
Processor Address Space :
 ras10_low            : 0x00000000 (b/s 0x00000000)
 ras10_high           : 0x07000000 (b/s 0x00000007)
 ras32_low            : 0x08000000 (b/s 0x00000008)
 ras32_high           : 0x0F000000 (b/s 0x0000000F)
 cs20_low             : 0xD0000000 (b/s 0x000000D0)
 cs20_high            : 0x74000000 (b/s 0x00000074)
 cs3_boot_low         : 0xF8000000 (b/s 0x000000F8)
 cs3_boot_high        : 0x7E000000 (b/s 0x0000007E)
 pci_io_low           : 0x00080000 (b/s 0x00000800)
 pci_io_high          : 0x00000000 (b/s 0x00000000)
 pci_mem_low          : 0x00020000 (b/s 0x00000200)
 pci_mem_high         : 0x7F000000 (b/s 0x0000007F)
 internal_spc_decode  : 0xA0000000 (b/s 0x000000A0)
 bus_err_low          : 0x00000000 (b/s 0x00000000)
 bus_err_high         : 0x00000000 (b/s 0x00000000)
.
.
.
```

show hardware

To display the hardware-specific information for a router, use the **show hardware**command in user EXEC or privileged EXEC mode.

show hardware

Syntax Description

This command has no arguments or keywords.

Command Modes

User EXEC (>) Privileged EXEC (#)

Command History

Release	Modification
12.4(22)T	This command was introduced.

Usage Guidelines

Use the **show hardware** command to display the hardware specific information for a router.

Examples

The following is sample output from the **show hardware** command:

```
Router# show hardware
Cisco IOS Software, 7200 Software (C7200-ADVENTERPRISEK9-M), Version 12.4(22)T,)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Fri 10-Oct-08 10:10 by prod_rel_team
ROM: System Bootstrap, Version 12.2(4r)B2, RELEASE SOFTWARE (fc2)
BOOTLDR: 7200 Software (C7200-KBOOT-M), Version 12.3(16), RELEASE SOFTWARE (fc4)
Router uptime is 1 day, 16 hours, 32 minutes
System returned to ROM by reload at 04:13:23 UTC Wed Aug 12 2009
System image file is "disk0:Default-IOS-Image-Do-Not-Delete"
Last reload reason: Reload Command
This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.
A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html
If you require further assistance please contact us by sending email to
export@cisco.com.
Cisco 7206VXR (NPE400) processor (revision A) with 491520K/32768K bytes of memo.
Processor board ID 31410931
R7000 CPU at 350MHz, Implementation 39, Rev 3.3, 256KB L2 Cache
6 slot VXR midplane, Version 2.7
Last reset from power-on
PCI bus mb0_mb1 (Slots 0, 1, 3 and 5) has a capacity of 600 bandwidth points.
Current configuration on bus mb0_mb1 has a total of 600 bandwidth points.
This configuration is within the PCI bus capacity and is supported.
PCI bus mb2 (Slots 2, 4, 6) has a capacity of 600 bandwidth points.
Current configuration on bus mb2 has a total of 180 bandwidth points
This configuration is within the PCI bus capacity and is supported.

Please refer to the following document "Cisco 7200 Series Port Adaptor
Hardware Configuration Guidelines" on Cisco.com <http://www.cisco.com>
for c7200 bandwidth points oversubscription and usage guidelines.
```

```
2 FastEthernet interfaces
4 Serial interfaces
125K bytes of NVRAM.
62976K bytes of ATA PCMCIA card at slot 0 (Sector size 512 bytes).
125440K bytes of ATA PCMCIA card at slot 1 (Sector size 512 bytes).
8192K bytes of Flash internal SIMM (Sector size 256K).
Configuration register is 0x2002
```

Related Commands

Command	Description
show interfaces	Displays statistics for all interfaces configured on the router or access server.

show health-monitor

To display the system Health Monitor status information, use the **show health-monitor**command in user EXEC or privileged EXEC mode.

show health-monitor [**summary**]

Syntax Description

summary	(Optional) Displays a summary of the status information.

Command Modes

User EXEC (>) Privileged EXEC (#)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.

Usage Guidelines

Use this command to display the state of the hardware and software subsystem. Health Monitor is a Cisco IOS subsystem that monitors the state of the individual hardware and software subsystems. This monitoring helps in early detection and recovery of faults in the subsystem.

Examples

The following is sample output from **show health-monitor**command. The fields are self explanatory.

```
Router# show health-monitor summary
Chassis:
   Power Supply                          Failure
   Temperature                                OK
   Fans                                       OK
Memory:
   Free Memory processor                      OK
   Memory Fragmentation Processor             OK
   Free Memory I/O                            OK
   Memory Fragmentation I/O                   OK
DFC's:
   Slot 1 - Empty DFC           Not in operation
   Slot 2 - Empty DFC           Not in operation
   Slot 3 - AS5X-FC                           OK
   Slot 4 - Empty DFC           Not in operation
   Slot 5 - Empty DFC           Not in operation
   Slot 6 - Empty DFC           Not in operation
   Slot 7 - Empty DFC           Not in operation
```

show history

To list the commands you have entered in the current EXEC session, use the **show history** command in EXEC mode.

show history

Syntax Description

This command has no arguments or keywords.

Command Modes

EXEC

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The command history feature provides a record of EXEC commands you have entered. The number of commands that the history buffer will record is determined by the **history size** line configuration command or the **terminal history size** EXEC command.

The table below lists the keys and functions you can use to recall commands from the command history buffer.

Table 1: History Keys

Key	Function
Ctrl-P or Up Arrow[1]	Recalls commands in the history buffer in a backward sequence, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Ctrl-N or Down Arrow1	Returns to more recent commands in the history buffer after recalling commands with Ctrl-P or the Up Arrow. Repeat the key sequence to recall successively more recent commands.

[1] The arrow keys function only with ANSI-compatible terminals.

Examples

The following is sample output from the **show history** command, which lists the commands the user has entered in EXEC mode for this session:

```
Router# show history
  help
  where
  show hosts
  show history
Router#
```

Related Commands

Command	Description
history size	Enables the command history function, or changes the command history buffer size for a particular line.

Command	Description
terminal history size	Enables the command history feature for the current terminal session, or changes the size of the command history buffer for the current terminal session.

show history all

To display command history and reload information of a router, use the **show history all**command in user EXEC or privileged EXEC mode.

show history all

Syntax Description

This command has no arguments or keywords.

Command Modes

User EXEC (>) Privileged EXEC (#)

Command History

Release	Modification
12.4(22)T	This command was introduced.

Usage Guidelines

Use the **show history all**command to display command history and reload information of a router.

Examples

The following is sample output from the **show history all**command:

```
Router# show history all
This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.
A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html
If you require further assistance please contact us by sending email to
export@cisco.com.
Cisco 7206VXR (NPE400) processor (revision A) with 491520K/32768K bytes of memo.
Processor board ID 31410931
R7000 CPU at 350MHz, Implementation 39, Rev 3.3, 256KB L2, 4096KB L3 Cache
6 slot VXR midplane, Version 2.7
Last reset from power-on
PCI bus mb0_mb1 (Slots 0, 1, 3 and 5) has a capacity of 600 bandwidth points.
Current configuration on bus mb0_mb1 has a total of 600 bandwidth points.
This configuration is within the PCI bus capacity and is supported.
PCI bus mb2 (Slots 2, 4, 6) has a capacity of 600 bandwidth points.
Current configuration on bus mb2 has a total of 180 bandwidth points
This configuration is within the PCI bus capacity and is supported.
Please refer to the following document "Cisco 7200 Series Port Adaptor
Hardware Configuration Guidelines" on Cisco.com <http://www.cisco.com>
for c7200 bandwidth points oversubscription and usage guidelines.
2 FastEthernet interfaces
4 Serial interfaces
125K bytes of NVRAM.
Installed image archive
```

```
*Aug 12 04:17:08.415: %LINEPROTO-5-UPDOWN: Line protocol on Interface VoIP-Nullp
*Aug 12 04:17:08.419: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state p
*Aug 12 04:17:08.419: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state p
*Aug 12 04:17:08.419: %LINK-3-UPDOWN: Interface Serial2/0, changed state to down
*Aug 12 04:17:08.419: %LINK-3-UPDOWN: Interface Serial2/1, changed state to down
*Aug 12 04:17:08.419: %LINK-3-UPDOWN: Interface Serial3/0, changed state to up
*Aug 12 04:17:08.419: %LINK-3-UPDOWN: Interface Serial3/1, changed state to up
*Aug 12 04:17:08.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface SSLVPN-VIp
62976K bytes of ATA PCMCIA card at slot 0 (Sector size 512 bytes).
125440K bytes of ATA PCMCIA card at slot 1 (Sector size 512 bytes).
8192K bytes of Flash internal SIMM (Sector size 256K).
*Aug 12 04:17:09.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEtherp
*Aug 12 04:17:09.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEtherp
*Aug 12 04:17:09.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0n
*Aug 12 04:17:09.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/1n
*Aug 12 04:17:09.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0p
*Aug 12 04:17:09.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/1p
*Aug 12 04:17:12.411: %LINK-3-UPDOWN: Interface Serial3/0, changed state to down
*Aug 12 04:17:12.411: %LINK-3-UPDOWN: Interface Serial3/1, changed state to down
*Aug 12 04:17:13.411: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0n
*Aug 12 04:17:13.411: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/1n
         --- System Configuration Dialog ---
Would you like to enter the initial configuration dialog? [yes/no]:
% Please answer 'yes' or 'no'.
Would you like to enter the initial configuration dialog? [yes/no]: no
Would you like to terminate autoinstall? [yes]: yes
CMD: 'access-list 199 permit icmp host 10.10.10.10 host 20.20.20.20' 04:18:15 U9
CMD: 'crypto map NiStTeSt1 10 ipsec-manual' 04:18:15 UTC Wed Aug 12 2009
CMD: 'match address 199
' 04:18:15 UTC Wed Aug 12 2009
CMD: 'set peer 20.20.20.20
' 04:18:15 UTC Wed Aug 12 2009
CMD: 'exit' 04:18:15 UTC Wed Aug 12 2009
CMD: 'no access-list 199' 04:18:15 UTC Wed Aug 12 2009
CMD: 'no crypto map NiStTeSt1' 04:18:15 UTC Wed Aug 12 2009
*Aug 12 04:18:15.403: %SYS-5-RESTART: System restarted --
Cisco IOS Software, 7200 Software (C7200-ADVENTERPRISEK9-M), Version 12.4(22)T,)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Fri 10-Oct-08 10:10 by prod_rel_team
*Aug 12 04:18:15.415: %ENTITY_ALARM-6-INFO: ASSERT INFO Fa0/0 Physical Port Adm
*Aug 12 04:18:15.415: %ENTITY_ALARM-6-INFO: ASSERT INFO Fa0/1 Physical Port Adm
*Aug 12 04:18:15.499: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
*Aug 12 04:18:15.499: %CRYPTO-6-GDOI_ON_OFF: GDOI is OFF
*Aug 12 04:18:15.599: %ENTITY_ALARM-6-INFO: ASSERT INFO Se2/0 Physical Port Adm
*Aug 12 04:18:15.599: %ENTITY_ALARM-6-INFO: ASSERT INFO Se2/1 Physical Port Adm
*Aug 12 04:18:15.599: %ENTITY_ALARM-6-INFO: ASSERT INFO Se3/0 Physical Port Adm
*Aug 12 04:18:15.599: %ENTITY_ALARM-6-INFO: ASSERT INFO Se3/1 Physical Port Adm
*Aug 12 04:18:15.599: %SNMP-5-COLDSTART: SNMP agent on host Router is undergoint
*Aug 12 04:18:15.823: %SYS-6-BOOTTIME: Time taken to reboot after reload =  314s
*Aug 12 04:18:16.715: %LINK-5-CHANGED: Interface Serial2/0, changed state to adn
*Aug 12 04:18:16.719: %LINK-5-CHANGED: Interface FastEthernet0/0, changed staten
*Aug 12 04:18:16.723: %LINK-5-CHANGED: Interface FastEthernet0/1, changed staten
*Aug 12 04:18:16.727: %LINK-5-CHANGED: Interface Serial2/1, changed state to adn
*Aug 12 04:18:16.727: %LINK-5-CHANGED: Interface Serial3/0, changed state to adn
*Aug 12 04:18:16.727: %LINK-5-CHANGED: Interface Serial3/1, changed state to adn
*Aug 12 04:18:17.719: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
*Aug 12 04:18:17.723: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEther9
CMD: 'conf t' 04:18:30 UTC Wed Aug 12 2009
CMD: 'hostname 7206-3' 04:19:02 UTC Wed Aug 12 2009
CMD: 'ip host sjc-tftp02 171.69.17.17' 04:19:02 UTC Wed Aug 12 2009
CMD: 'ip host sjc-tftp01 171.69.17.19' 04:19:03 UTC Wed Aug 12 2009
CMD: 'ip host dirt 171.69.1.129' 04:19:03 UTC Wed Aug 12 2009
CMD: 'interface FastEthernet0/0' 04:19:03 UTC Wed Aug 12 2009
```

```
CMD: 'no ip proxy-arp' 04:19:03 UTC Wed Aug 12 2009
CMD: 'ip address 10.4.9.80 255.255.255.0' 04:19:03 UTC Wed Aug 12 2009
CMD: 'no shutdown' 04:19:04 UTC Wed Aug 12 2009
CMD: 'exit' 04:19:04 UTC Wed Aug 12 2009
CMD: 'ip classless' 04:19:05 UTC Wed Aug 12 2009
*Aug 12 04:19:06.123: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state p
*Aug 12 04:19:06.123: %ENTITY_ALARM-6-INFO: CLEAR INFO Fa0/0 Physical Port Admi9
CMD: 'ip default-network 0.0.0.0' 04:19:06 UTC Wed Aug 12 2009
CMD: 'ip default-gateway 10.4.9.1' 04:19:06 UTC Wed Aug 12 2009
CMD: 'config-register 0x2002' 04:19:07 UTC Wed Aug 12 2009
```

Related Commands

Command	Description
show history	Displays commands entered in the current EXEC session.

show hosts

To display the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses specific to a particular Domain Name System (DNS) view or for all configured DNS views, use the **show hosts** command in privileged EXEC mode.

show hosts [**vrf** *vrf-name*] [{**view** [{*view-name* | **default**}]}] [**all**] [{*hostname* | **summary**}]

Syntax Description

vrf *vrf-name*	(Optional) The *vrf-name* argument specifies the name of the Virtual Private Network (VPN) routing and forwarding (VRF) instance associated with the DNS view whose hostname cache entries are to be displayed. Default is the global VRF (that is, the VRF whose name is a NULL string) with the specified or default DNS view. **Note** More than one DNS view can be associated with a VRF. To uniquely identify a DNS view, specify both the view name and the VRF with which it is associated.
view *view-name*	(Optional) The *view-name* argument specifies the DNS view whose hostname cache information is to be displayed. Default is the default (unnamed) DNS view associated with the specified or global VRF. **Note** More than one DNS view can be associated with a VRF. To uniquely identify a DNS view, specify both the view name and the VRF with which it is associated.
default	(Optional) Displays the default view.
all	(Optional) Display all the host tables.
hostname	(Optional) The specified hostname cache information displayed is to be limited to entries for a particular hostname. Default is the hostname cache information for all hostname entries in the cache.
summary	(Optional) The specified hostname cache information is to be displayed in brief summary format. Disabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
10.0	This command was introduced.
12.2T	Support was added for Cisco modem user interface feature.
12.4(4)T	The **vrf**, **all**, and **summary** keywords and *vrf-name* and *hostname* arguments were added.
12.4(9)T	The **view** keyword and *view-name* argument were added.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

This command displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses specific to a particular DNS view or for all configured DNS views.

If you specify the **show hosts** command without any optional keywords or arguments, only the entries in the global hostname cache will be displayed.

If the output from this command extends beyond the bottom of the screen, press the Space bar to continue or press the Q key to terminate command output.

Examples

The following is sample output from the **show hosts** command with no parameters specified:

```
Router# show hosts

Default domain is CISCO.COM
Name/address lookup uses domain service
Name servers are 192.0.2.220
Host Flag Age Type Address(es)
EXAMPLE1.CISCO.COM (temp, OK) 1 IP 192.0.2.10
EXAMPLE2.CISCO.COM (temp, OK) 8 IP 192.0.2.50
EXAMPLE3.CISCO.COM (temp, OK) 8 IP 192.0.2.115
EXAMPLE4.CISCO.COM (temp, EX) 8 IP 192.0.2.111
EXAMPLE5.CISCO.COM (temp, EX) 0 IP 192.0.2.27
EXAMPLE6.CISCO.COM (temp, EX) 24 IP 192.0.2.30
```

The following is sample output from the **show hosts** command that specifies the VRF vpn101:

```
Router# show hosts vrf vpn101

Default domain is example.com
Domain list: example1.com, example2.com, example3.com
Name/address lookup uses domain service
Name servers are 192.0.2.204, 192.0.2.205, 192.0.2.206
Codes: UN - unknown, EX - expired, OK - OK, ?? - revalidate
       temp - temporary, perm - permanent
       NA - Not Applicable None - Not defined
Host                      Port  Flags      Age Type   Address(es)
user                      None  (perm, OK)  0   IP    192.0.2.001
www.example.com           None  (perm, OK)  0   IP    192.0.2.111
                                                      192.0.2.112
```

The table below describes the significant fields shown in the display.

Table 2: show hosts Field Descriptions

Field	Description
Default domain	Default domain name to be used to complete unqualified names if no domain list is defined.
Domain list	List of default domain names to be tried in turn to complete unqualified names.
Name/address lookup	Style of name lookup service.
Name servers	List of name server hosts.
Host	Learned or statically defined hostname. Statically defined hostname-to-address mappings can be added to the DNS hostname cache for a DNS view by using the **ip hosts** command.
Port	TCP port number to connect to when using the defined hostname in conjunction with an EXEC connect or Telnet command.
Flags	Indicates additional information about the hostname-to-IP address mapping. Possible values are as follows: • EX--Entries marked EX are expired. • OK--Entries marked OK are believed to be valid. • perm--A permanent entry is entered by a configuration command and is not timed out. • temp--A temporary entry is entered by a name server; the Cisco IOS software removes the entry after 72 hours of inactivity. • ??--Entries marked ?? are considered suspect and subject to revalidation.
Age	Number of hours since the software last referred to the cache entry.
Type	Type of address. For example, IP, Connectionless Network Service (CLNS), or X.121. If you have used the **ip hp-host global** configuration command, the **show hosts** command will display these hostnames as type HP-IP.
Address(es)	IP address of the host. One host may have up to eight addresses.

Related Commands

Command	Description
clear host	Removes static hostname-to-address mappings from the hostname cache for the specified DNS view or all DNS views.
ip host	Defines static hostname-to-address mappings in the DNS hostname cache for a DNS view.

show html

To display module and port information, use the **show html** command in privileged EXEC mode.

show html {**module** [**ports** [**l2**]] | **port** [{**all** | **l2** | **l3**}] [**shortnames**]} {**command** *line* | **count** | **names** | **options**}

Syntax Description

module	Displays module information.
ports	(Optional) Displays the number of ports on the module.
l2	(Optional) Displays information about the Layer2 (l2) module.
port	Displays port information.
all	(Optional)Displays information about the Layer 2 and Layer 3 modules.
l2	(Optional) Displays information about the Layer2 (l2) module.
l3	(Optional) Displays information about the Layer3 (l3) module.
shortnames	(Optional) Displays port short names.
command	Displays execute command over ports information.
line	Displays command to execute over modules information.
count	Displays the module count.
names	Displays the module names.
options	Displays the module options.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(24)T	This command was introduced in a release earlier than Cisco IOS Release 12.4(24)T.
12.2(33)SXI	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SXI.
12.2(33)SRC	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SRC.

Usage Guidelines

Use the **show html**command to display module and port information.

Examples

The following is sample output from the **show html** command using the **port** and **names**keywords. The field descriptions are self-explanatory.

```
Router# show html port names
this[0] = "FastEthernet0/0";
this[1] = "FastEthernet0/1";
this[2] = "Serial2/0";
this[3] = "Serial2/1";
```

```
this[4] = "Serial3/0";
this[5] = "Serial3/0.1";
this[6] = "Serial3/1";
this[7] = "Tunnel0";
this[8] = "Tunnel1";
this[9] = "Tunnel2";
this[10] = "Tunnel3";
this[11] = "Virtual-Access1";
this[12] = "Virtual-Template1";
this[13] = "vmi1";
this[14] = "vmi2";
```

The following is sample output from the **show html** command using the **port**, **all, and options**keywords. The ouput is self-explanatory.

```
Router# show html port all options

<option>FastEthernet0/0
<option>FastEthernet0/1
<option>Serial2/0
<option>Serial2/1
<option>Serial3/0
<option>Serial3/0.1
<option>Serial3/1
<option>Tunnel0
<option>Tunnel1
<option>Tunnel2
<option>Tunnel3
<option>Virtual-Access1
<option>Virtual-Template1
<option>VoIP-Null0
<option>vmi1
<option>vmi2
```

show idb

To display information about the status of interface descriptor blocks (IDBs), use the **show idb**command in privileged EXEC mode.

show idb

Syntax Description

This command has nor arguments or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.1	This command was introduced.
12.2(15)T	The output of this command was changed to show additional information.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following is sample output from the **show idb** command:

```
Router# show idb
Maximum number of Software IDBs 8192.  In use 17.
                      HWIDBs     SWIDBs
Active                     5         14
Inactive                  10          3
Total IDBs                15         17
Size each (bytes)       5784       2576
Total bytes            86760      43792
HWIDB#1   1   2   GigabitEthernet0/0 0 5, HW IFINDEX, Ether)
HWIDB#2   2   3   GigabitEthernet9/0 0 5, HW IFINDEX, Ether)
HWIDB#3   3   4   GigabitEthernet9/1 6 5, HW IFINDEX, Ether)
HWIDB#4   4   5   GigabitEthernet9/2 6 5, HW IFINDEX, Ether)
HWIDB#5   13  1   Ethernet0 4 5, HW IFINDEX, Ether)
```

The table below describes the significant fields shown in the display.

Table 3: show idb Field Descriptions

Field	Description
In use	Total number of software IDBs (SWIDBs) that have been allocated. This number never decreases. SWIDBs are never deallocated.
Active	Total number of hardware IDBs (HWIDBs) and SWIDBs that are allocated and in use.
Inactive	Total number of HWIDBs and SWIDBs that are allocated but not in use.
Total	Total number of HWIDBs and SWIDBs that are allocated.

show idprom

To display the identification programmable read-only memory (IDPROM) information for field-replaceable units (FRUs), use the **show idprom** command in privileged EXEC mode.

show idprom {**all***frutype*} [**detail**]

Syntax Description

all	Displays the information for all FRU types.
frutype	Type of FRU for information to be displayed; see the “Usage Guidelines” section for valid values.
detail	(Optional) Displays the detailed display of IDPROM data (verbose).

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(14)SX	This command was introduced on the Supervisor Engine 720.
12.2(17d)SXB	Support for this command on the Supervisor Engine 2 was integrated into Release 12.2(17d)SXB.

Release	Modification
12.2(18)SXE	The **module** keyword was modified to support slot/subslot addressing for shared port adapters (SPAs) and SPA interface processors (SIPs), and the optional **clei** keyword was added. The **interface** keyword was replaced by the **transceiver** keyword.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Valid entries for *frutype* are as follows:

- **backplane**
- **clock** *number* --1 and 2.
- **earl** *slot* --See the following paragraph for valid slot values.
- **module** *slot* / *port* | *slot* | *slot* / *subslot*[**clei**] }--See the following paragraphs for valid values and descriptions.
- **rp** *slot* --See the following paragraph for valid slot values.
- **power-supply** --1 and 2.
- **supervisor** *slot* --See the following paragraph for valid slot values.
- **transceiver** *slot* / *subslot* / *port* | *slot* / *subslot* **GigabitEthernet** | **GigabitEthernetWAN**]}
- **vtt** *number* --1 to 3.

The **module** *slot*/*port* argument designates the module slot location and port number.

Valid values for *slot* depend on the specified interface type and the chassis and module that are used. For example, if you specify a Gigabit Ethernet interface and have a 48-port 10/100BASE-T Ethernet module that is installed in a 13-slot chassis, valid values for the module number are from 1 to 13 and valid values for the port number are from 1 to 48.

The **module** {*slot* | *slot*/*subslot* [**clei**]} syntax designates either the *slot*location alone of theSIP in the chassis (to show information for the SIP only), or the *slot*location of theSIP and the *subslot* location of a SPA installed within the SIP (to display information for a SPA only). Valid values for *slot* depend on the chassis model (2-13), and valid values for *subslot* depend on the SIP type (such as 0-3 for a Cisco 7600 SIP-200 and Cisco 7600 SIP-400). The optional **clei** keyword specifies display of the Common Language Equipment Identification (CLEI) information for the specified SIP or SPA.

Use the **show idprom backplane** command to display the chassis serial number.

Use the **transceiver** *slot* / *subslot* / *port* form of the command to display information for transceivers installed in a SPA, where *slot* designates the location of the SIP, *subslot* designates the location of the SPA, and *port* designates the interface number.

The **interface** *interface slot* keyword and arguments supported on GBIC security-enabled interfaces have been replaced by the **transceiver** keyword option.

To specify LAN Gigabit Ethernet interfaces, use the **show idprom transceiver***slot*/*subslot***GigabitEthernet** form of the command.

- To specify WAN Gigabit Ethernet interfaces, use the **show idprom transceiver***slot*/*subslot***GigabitEthernetWAN** form of the command.

Examples

This example shows how to display IDPROM information for clock 1:

```
Router#
show idprom clock 1
IDPROM for clock #1
  (FRU is 'Clock FRU')
  OEM String = 'Cisco Systems'
  Product Number = 'WS-C6000-CL'
  Serial Number = 'SMT03073115'
  Manufacturing Assembly Number = '73-3047-04'
  Manufacturing Assembly Revision = 'A0'
  Hardware Revision = 1.0
  Current supplied (+) or consumed (-) = 0.000A
```

The following table describes the significant fields shown in the display.

Table 4: show idprom Field Descriptions

Field	Description
FRU is	Indicates the type of the field-replacement unit (FRU) to which the information that follows applies.
OEM String	Names the original equipment manufacturer (OEM).
Product Number	A number that identifies a product line.
Serial Number	A number that uniquely identifies the product itself.
Manufacturing Assembly Number	A number that identifies the hardware identification number.
Manufacturing Assembly Revision	A number that identifies the manufacturing assembly number.
Hardware Revision	A number that represents the hardware upgrade.
Current supplied (+) or consumed (-)	Indicated the amount of electrical current that the device supples or uses.

This example shows how to display IDPROM information for power supply 1:

```
Router#
show idprom power-supply 1
IDPROM for power-supply #1
  (FRU is '110/220v AC power supply, 1360 watt')
  OEM String = 'Cisco Systems, Inc.'
  Product Number = 'WS-CAC-1300W'
  Serial Number = 'ACP03020001'
  Manufacturing Assembly Number = '34-0918-01'
  Manufacturing Assembly Revision = 'A0'
  Hardware Revision = 1.0
  Current supplied (+) or consumed (-) = 27.460A
```

This example shows how to display detailed IDPROM information for power supply 1:

```
Router#
show idprom power-supply 1 detail
IDPROM for power-supply #1
IDPROM image:
```

```
  (FRU is '110/220v AC power supply, 1360 watt')
IDPROM image block #0:
  hexadecimal contents of block:
  00: AB AB 01 90 11 BE 01 00 00 02 AB 01 00 01 43 69    ..............Ci
  10: 73 63 6F 20 53 79 73 74 65 6D 73 2C 20 49 6E 63    sco Systems, Inc
  20: 2E 00 57 53 2D 43 41 43 2D 31 33 30 30 57 00 00    ..WS-CAC-1300W..
  30: 00 00 00 00 00 00 41 43 50 30 33 30 32 30 30 30    ......ACP0302000
  40: 31 00 00 00 00 00 00 00 00 00 33 34 2D 30 39 31    1.........34-091
  50: 38 2D 30 31 00 00 00 00 00 00 41 30 00 00 00 00    8-01......A0....
  60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
  70: 00 00 00 01 00 00 00 00 00 00 00 09 00 0C 00 03    ................
  80: 00 01 00 06 00 01 00 00 00 00 0A BA 00 00 00 00    ................
  block-signature = 0xABAB, block-version = 1,
  block-length = 144, block-checksum = 4542
  *** common-block ***
  IDPROM capacity (bytes) = 256  IDPROM block-count = 2
  FRU type = (0xAB01,1)
  OEM String = 'Cisco Systems, Inc.'
  Product Number = 'WS-CAC-1300W'
  Serial Number = 'ACP03020001'
  Manufacturing Assembly Number = '34-0918-01'
  Manufacturing Assembly Revision = 'A0'
  Hardware Revision = 1.0
  Manufacturing bits = 0x0  Engineering bits = 0x0
  SNMP OID = 9.12.3.1.6.1.0
  Power Consumption = 2746 centiamperes    RMA failure code = 0-0-0-0
  *** end of common block ***
IDPROM image block #1:
  hexadecimal contents of block:
  00: AB 01 01 14 02 5F 00 00 00 00 00 00 00 00 0A BA    ....._..........
  10: 0A BA 00 16                                        ....
  block-signature = 0xAB01, block-version = 1,
  block-length = 20, block-checksum = 607
  *** power supply block ***
  feature-bits:  00000000 00000000
  rated current at 110v:  2746    rated current at 220v:  2746   (centiamperes)
  CISCO-STACK-MIB SNMP OID = 22  *** end of power supply block ***
End of IDPROM image
```

This example shows how to display IDPROM information for the backplane:

```
Router#
show idprom backplane
IDPROM for backplane #0
  (FRU is 'Catalyst 6000 9-slot backplane')
  OEM String = 'Cisco Systems'
  Product Number = 'WS-C6009'
  Serial Number = 'SCA030900JA'
  Manufacturing Assembly Number = '73-3046-04'
  Manufacturing Assembly Revision = 'A0'
  Hardware Revision = 1.0
  Current supplied (+) or consumed (-) = 0.000A
```

The following example shows sample output for a Cisco 7600 SIP-400 installed in slot 3 of the router:

```
Router# show idprom module 3
IDPROM for module #3
  (FRU is '4-subslot SPA Interface Processor-400')
  OEM String = 'Cisco Systems'
  Product Number = '7600-SIP-400'
  Serial Number = 'JAB0851042X'
  Manufacturing Assembly Number = '73-8404-10'
```

```
  Manufacturing Assembly Revision = '09'
  Hardware Revision = 0.95
  Current supplied (+) or consumed (-) = -6.31A
```

The following example shows sample output for the **clei** form of the command on a Cisco 7600 SIP-200 installed in slot 2 of the router:

```
Router# show idprom module 2 clei
FRU             PID                 VID SN          CLEI
--------------- ------------------- --- ----------- ----------
module #2       7600-SIP-200        V01
```

The following example shows sample output for the **detail** form of the command on a Cisco 7600 SIP-400 installed in slot 3 of the router:

```
Router# show idprom module 3 detail
IDPROM for module #3
IDPROM image:
  (FRU is '4-subslot SPA Interface Processor-400')
IDPROM image block #0:
  block-signature = 0xABAB, block-version = 3,
  block-length = 160, block-checksum = 4600
  *** common-block ***
  IDPROM capacity (bytes) = 512  IDPROM block-count = 2
  FRU type = (0x6003,1103)
  OEM String = 'Cisco Systems'
  Product Number = '7600-SIP-400'
  Serial Number = 'JAB0851042X'
  Manufacturing Assembly Number = '73-8404-10'
  Manufacturing Assembly Revision = '09'
  Manufacturing Assembly Deviation = '00'
  Hardware Revision = 0.95
  Manufacturing bits = 0x0  Engineering bits = 0x0
  SNMP OID = 9.5.1.3.1.1.2.1103
  Power Consumption = -631 centiamperes    RMA failure code = 0-0-0-0
  CLEI =
  VID =
  *** end of common block ***
IDPROM image block #1:
  block-signature = 0x6003, block-version = 2,
  block-length = 103, block-checksum = 2556
  *** linecard specific block ***
  feature-bits =   00000000 00000000
  hardware-changes-bits =   00000000 00000000
  card index = 158
  mac base = 0012.4310.D840
  mac_len = 128
  num_processors = 1
  epld_num = 0
  epld_versions = 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000
  port numbers:
    pair #0: type=00, count=00
    pair #1: type=00, count=00
    pair #2: type=00, count=00
    pair #3: type=00, count=00
    pair #4: type=00, count=00
    pair #5: type=00, count=00
    pair #6: type=00, count=00
    pair #7: type=00, count=00
  sram_size = 0
  sensor_thresholds =
    sensor #0: critical = 75 oC, warning = 60 oC
```

```
    sensor #1: critical = 70 oC, warning = 55 oC
    sensor #2: critical = 80 oC, warning = 65 oC
    sensor #3: critical = 75 oC, warning = 60 oC
    sensor #4: critical = -128 oC (sensor not present), warning = -128 oC (sensor not
present)
    sensor #5: critical = -128 oC (sensor not present), warning = -128 oC (sensor not
present)
    sensor #6: critical = -128 oC (sensor not present), warning = -128 oC (sensor not
present)
    sensor #7: critical = -128 oC (sensor not present), warning = -128 oC (sensor not
present)
  max_connector_power = 3600
  cooling_requirement = 35
  ambient_temp = 55
  *** end of linecard specific block ***

End of IDPROM image
```

The following example shows sample output for a 4-Port OC-3c/STM-1 ATM SPA installed in subslot 0 of the SIP installed in slot 5 of the router:

```
Router# show idprom module 5/0
IDPROM for SPA module #5/0
        (FRU is '4-port OC3/STM1 ATM Shared Port Adapter')
        Product Identifier (PID) : SPA-4XOC3-ATM
        Version Identifier (VID) : V01
        PCB Serial Number        : PRTA2604138
        Top Assy. Part Number    : 68-2177-01
        73/68 Board Revision     : 05
        73/68 Board Revision     : 01
        Hardware Revision        : 0.224
        CLEI Code                : UNASSIGNED
```

The following example shows sample output for the **clei** form of the command for a 4-Port OC-3c/STM-1 POS SPA installed in subslot 3 of the SIP installed in slot 2 of the router:

```
Router# show idprom module 2/3 clei
FRU             PID                  VID SN          CLEI
--------------- -------------------- --- ----------- ----------
SPA module #2/3 SPA-4XOC3-POS        V01 PRTA0304155 UNASSIGNED
```

The following example shows sample output for the **detail** form of the command for a 4-Port OC-3c/STM-1 POS SPA installed in subslot 3 of the SIP installed in slot 2 of the router:

```
Router# show idprom module 2/3 detail
IDPROM for SPA module #2/3
        (FRU is '4-port OC3/STM1 POS Shared Port Adapter')
        EEPROM version           : 4
        Compatible Type          : 0xFF
        Controller Type          : 1088
        Hardware Revision        : 0.230
        Boot Timeout             : 0 msecs
        PCB Serial Number        : PRTA0304155
        Part Number              : 73-9313-02
        73/68 Board Revision     : 04
        Fab Version              : 02
        RMA Test History         : 00
        RMA Number               : 0-0-0-0
        RMA History              : 00
        Deviation Number         : 0
        Product Identifier (PID) : SPA-4XOC3-POS
        Version Identifier (VID) : V01
```

```
Top Assy. Part Number    : 68-2169-01
73/68 Board Revision     : 10
System Clock Frequency   : 00 00 00 00 00 00 00 00
                           00 00 00 00 00
CLEI Code                : UNASSIGNED
Base MAC Address         : 00 00 00 00 00 00
MAC Address block size   : 0
Manufacturing Test Data  : 00 00 00 00 00 00 00 00
Field Diagnostics Data   : 00 00 00 00 00 00 00 00
Calibration Data         : Minimum: 0 dBmV, Maximum: 0 dBmV
      Calibration values :
Power Consumption        : 16200 mWatts (Maximum)
Environment Monitor Data : 01 08 F6 48 43 34 F6 48
                           43 34 02 31 0C E4 46 32
                           28 13 07 09 C4 46 32 28
                           13 07 00 00 00 00 00 00
                           00 05 DC 46 32 28 13 07
                           00 00 00 00 00 00 00 00
                           00 00 00 00 00 00 00 00
                           00 00 00 00 00 FE 02 00
                           00
Asset ID                 :
Asset Alias              :
```

show inventory

To display the product inventory listing of all Cisco products installed in the networking device, use the **show inventory** command in user EXEC or privileged EXEC mode.

show inventory [**raw**] [**entity**]

Syntax Description

raw	(Optional) Retrieves information about all of the Cisco products--referred to as entities--installed in the Cisco networking device, even if the entities do not have a product ID (PID) value, a unique device identifier (UDI), or other physical identification.
entity	(Optional) Name of a Cisco entity (for example, chassis, backplane, module, or slot). A quoted string may be used to display very specific UDI information; for example “sfslot 1” will display the UDI information for slot 1 of an entity named sfslot.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.0(27)S	This command was integrated into Cisco IOS Release 12.0(27)S.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2(27)SBC	This command was integrated into Cisco IOS Release 12.2(27)SBC.
12.2(18)SXE5	This command was integrated into Cisco IOS Release 12.2(18)SXE5.

Usage Guidelines

The **show inventory** command retrieves and displays inventory information about each Cisco product in the form of a UDI. The UDI is a combination of three separate data elements: a product identifier (PID), a version identifier (VID), and the serial number (SN).

The PID is the name by which the product can be ordered; it has been historically called the “Product Name” or “Part Number.” This is the identifier that one would use to order an exact replacement part.

The VID is the version of the product. Whenever a product has been revised, the VID will be incremented. The VID is incremented according to a rigorous process derived from Telcordia GR-209-CORE, an industry guideline that governs product change notices.

The SN is the vendor-unique serialization of the product. Each manufactured product will carry a unique serial number assigned at the factory, which cannot be changed in the field. This is the means by which to identify an individual, specific instance of a product.

The UDI refers to each product as an entity. Some entities, such as a chassis, will have subentities like slots. Each entity will display on a separate line in a logically ordered presentation that is arranged hierarchically by Cisco entities.

Use the **show inventory** command without options to display a list of Cisco entities installed in the networking device that are assigned a PID.

Examples

The following is sample output from the **show inventory**command without any keywords or arguments. This sample output displays a list of Cisco entities installed in a router that are assigned a PID.

```
Router# show inventory
NAME: “Chassis”, DESCR: “12008/GRP chassis”
PID: GSR8/40           ,  VID: V01,  SN: 63915640
NAME: “slot 0”, DESCR: “GRP”
PID: GRP-B             ,  VID: V01,  SN: CAB021300R5
NAME: “slot 1”, DESCR: “4 port ATM OC3 multimode”
PID: 4OC3/ATM-MM-SC    ,  VID: V01,  SN: CAB04036GT1
NAME: “slot 3”, DESCR: “4 port 0C3 POS multimode”
PID: LC-4OC3/POS-MM    ,  VID: V01,  SN: CAB014900GU
NAME: “slot 5”, DESCR: “1 port Gigabit Ethernet”
PID: GE-GBIC-SC-B      ,  VID: V01,  SN: CAB034251NX
NAME: “slot 7”, DESCR: “GRP”
PID: GRP-B             ,  VID: V01,  SN: CAB0428AN4O
NAME: “slot 16”, DESCR: “GSR 12008 Clock Scheduler Card”
PID: GSR8-CSC/ALRM     ,  VID: V01,  SN: CAB0429AUYH
NAME: “sfslot 1”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0428ALOS
NAME: “sfslot 2”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0429AU0M
NAME: “sfslot 3”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0429ARD7
NAME: “PSslot 1”, DESCR: “GSR 12008 AC Power Supply”
PID: FWR-GSR8-AC-B     ,  VID: V01,  SN: CAB041999CW
```

The table below describes the fields shown in the display.

Table 5: show inventory Field Descriptions

Field	Description
NAME	Physical name (text string) assigned to the Cisco entity. For example, console or a simple component number (port or module number), such as "1," depending on the physical component naming syntax of the device.
DESCR	Physical description of the Cisco entity that characterizes the object. The physical description includes the hardware serial number and the hardware revision.
PID	Entity product identifier. Equivalent to the entPhysicalModelName MIB variable in RFC 2737.
VID	Entity version identifier. Equivalent to the entPhysicalHardwareRev MIB variable in RFC 2737.
SN	Entity serial number. Equivalent to the entPhysicalSerialNum MIB variable in RFC 2737.

For diagnostic purposes, the **show inventory**command can be used with the **raw** keyword to display every RFC 2737 entity including those without a PID, UDI, or other physical identification.

Note The **raw** keyword option is primarily intended for troubleshooting problems with the **show inventory** command itself.

```
Router# show inventory raw
NAME: "Chassis", DESCR: "12008/GRP chassis"
PID:                    ,  VID: V01,  SN: 63915640
NAME: "slot 0", DESCR: "GRP"
PID:                    ,  VID: V01,  SN: CAB021300R5
NAME: "slot 1", DESCR: "4 port ATM OC3 multimode"
PID: 4OC3/ATM-MM-SC     ,  VID: V01,  SN: CAB04036GT1
NAME: "slot 3", DESCR: "4 port 0C3 POS multimode"
PID: LC-4OC3/POS-MM     ,  VID: V01,  SN: CAB014900GU
```

Enter the **show inventory**command with an *entity* argument value to display the UDI information for a specific type of Cisco entity installed in the networking device. In this example, a list of Cisco entities that match the sfslot argument string is displayed.

```
Router# show inventory sfslot
NAME: "sfslot 1", DESCR: "GSR 12008 Switch Fabric Card"
PID: GSR8-SFC           ,  VID: V01,  SN: CAB0428ALOS
NAME: "sfslot 2", DESCR: "GSR 12008 Switch Fabric Card"
PID: GSR8-SFC           ,  VID: V01,  SN: CAB0429AU0M
NAME: "sfslot 3", DESCR: "GSR 12008 Switch Fabric Card"
PID: GSR8-SFC           ,  VID: V01,  SN: CAB0429ARD7
```

You can request even more specific UDI information using the **show inventory**command with an *entity* argument value that is enclosed in quotation marks. In this example, only the details for the entity that exactly matches the sfslot 1 argument string are displayed.

```
Router# show inventory "sfslot 1"
NAME: "sfslot 1", DESCR: "GSR 12008 Switch Fabric Card"
PID: GSR8-SFC           ,  VID: V01,  SN: CAB0428ALOS
```

Related Commands

Command	Description
show diag	Displays diagnostic information about the controller, interface processor, and port adapters for a networking device.
show tech-support	Displays general information about the router when it reports a problem.

show location

To display the location information for an endpoint, use the **show location** command in user EXEC or privileged EXEC mode.

show location {{**civic-location** | **custom-location** | **geo-location** {**identifier** *id* | **interface** *name type* | **static**}} | **host**}

Syntax Description

civic-location	Specifies the civic location information.
custom-location	Specifies the custom location information.
geo-location	Specifies the geo-spatial location information.
host	Specifies the civic, custom, and geo-spatial host location information.
identifier *id*	Specifies the information identifier of the civic location, custom location, and geo-spatial location.
interface *type number*	Specifies the interface type and interface number.
static	Specifies the configured location information.

Command Modes

User EXEC (>)
Privileged EXEC (#)

Command History

Release	Modification
12.2(40)SE	This command was introduced.
12.2(55)SE	This command was modified. The output was enhanced to display location information obtained from Cisco Discovery Protocol.
15.1(1)SG	This command was modified. The **custom-location** and **geo-location** keywords were added.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.

Examples

The following sample output from the **show location civic-location** command displays all the civic location information for a specific identifier:

```
Device# show location civic-location identifier test
Civic location information
```

```
--------------------------
Identifier              : test
Building                : 24
City                    : Milpitas
State                   : California
Ports                   : Gi1/0/10
```

The following sample output from the **show location custom-location** command displays custom location information of a host device:

```
Device# show location custom-location identifier
Custom location information
---------------------------
Identifier: host
Name      : bgl15
Value     : IDF2.5
The following sample output from the show location geo-location
 command displays geo-spatial location information of a device:
Device# show location geo-location identifier apjtpk

Geo location information
------------------------
Identifier  : apjtpk
Latitude    : 54.45
Longitude   : 37.43
Altitude    : 5 floor
Resolution  : 54.45
The following sample output from the show location host
 command displays all host information of a device:
Device# show location host
Civic location information
--------------------------
Identifier              : host
County                  : raps
City Division           : SJ
Neighborhood            : lake
Street Group            : G2
Leading street direction: trav
Trailing street suffix  : C76
Street number           : 18
Street number suffix    : 54
Landmark                : park
Name                    : KMD
Building                : bgl13
Unit                    : apjtpk
Floor                   : 3
Room                    : Andaman
Type of place           : office
Postal community name   : ios
Post office box         : 12
Additional code         : apjtpk
Seat                    : B5-10
Primary road name       : outerringrd
Road section            : east
Branch road name        : venus
Sub branch road name    : Tata
Street name postmodifier: ret
City                    : Boston
State                   : CA
Postal code             : 1345
Additional location     : cauveri
Custom location information
---------------------------
```

```
Identifier: host
Name      : bgl15
Value     : IDF2.5
Geo location information
------------------------
Identifier  : host
Latitude    : 12.34
Longitude   : 56.78
```

The following table describes the significant fields shown in the display.

Table 6: show location Field Descriptions

Field	Description
Identifier	Information identifier of the civic location, custom location, and geo-spatial location.
Name	Name of the configured custom location identifier.
Value	Configured value of the custom location identifier.
Latitude	Configured latitude information of the device.
Longitude	Configured longitude information of the device.
Altitude	Configured altitude information of the device.
Resolution	Configured resolution for the latitude and longitude.

Related Commands

Command	Description
location civic-location identifier	Configures the civic location information of a device.
location custom-location identifier	Configures the custom location information of a device.
location geo-location identifier	Configures the geo-spatial location of a device such as latitude, longitude, altitude, and resolution.

show logging

To display the state of system logging (syslog) and the contents of the standard system logging buffer, use the **show logging** command in privileged EXEC mode.

show logging [{**slot** *slot-number* | **summary**}]

Syntax Description

slot *slot-number*	(Optional) Displays information in the syslog history table for a specific line card. Slot numbers range from 0 to 11 for the Cisco 12012 Internet router and from 0 to 7 for the Cisco 12008 Internet router.
summary	(Optional) Displays counts of messages by type for each line card.

Command Modes

Privileged EXEC

Command History

Release	Modification
10.0	This command was introduced.
11.2 GS	The **slot** and **summary** keywords were added for the Cisco 12000.
12.2(8)T	Command output was expanded to show the status of the logging count facility (“Count and time-stamp logging messages”).
12.2(15)T	Command output was expanded to show the status of XML syslog formatting.
12.3(2)T	Command output was expanded (on supported software images) to show details about the status of system logging processed through the Embedded Syslog Manager (ESM). These lines appear as references to “filtering” or “filter modules”.
12.3(2)XE	This command was integrated into Cisco IOS Release 12.3(2)XE.
12.2(14)SX	This command was integrated into Cisco IOS Release 12.2(14)SX.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(11)T	The CLI output was modified to show message discriminators defined at the router and syslog sessions associated with those message discriminators.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

Usage Guidelines

This command displays the state of syslog error and event logging, including host addresses, and which logging destinations (console, monitor, buffer, or host) logging is enabled. This command also displays Simple Network Management Protocol (SNMP) logging configuration parameters and protocol activity.

This command will display the contents of the standard system logging buffer, if logging to the buffer is enabled. Logging to the buffer is enabled or disabled using the [**no**] **logging buffered** command. The number of system error and debugging messages in the system logging buffer is determined by the configured size of the syslog buffer. This size of the syslog buffer is also set using the **logging buffered** command.

To enable and set the format for syslog message time stamping, use the **service timestamps log** command.

If debugging is enabled (using any **debug** command), and the logging buffer is configured to include level 7 (debugging) messages, debug output will be included in the system log. Debugging output is not formatted like system error messages and will not be preceded by the percent symbol (%).

Examples

The following is sample output from the **show logging** command on a software image that supports the Embedded Syslog Manager (ESM) feature:

```
Router# show logging

Syslog logging: enabled (10 messages dropped, 5 messages rate-limited,
                0 flushes, 0 overruns, xml disabled, filtering disabled)
    Console logging: level debugging, 31 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: disabled
    Buffer logging: level errors, 36 messages logged, xml disabled,
                    filtering disabled
```

```
    Logging Exception size (8192 bytes)
    Count and timestamp logging messages: disabled
No active filter modules.
    Trap logging: level informational, 45 message lines logged

Log Buffer (8192 bytes):
```

The following example shows output from the **show logging** command after a message discriminator has been configured. Included in this example is the command to configure the message discriminator.

```
c7200-3(config)# logging discriminator ATTFLTR1 severity includes 1,2,5 rate-limit 100

Specified MD by the name ATTFLTR1 is not found.
Adding new MD instance with specified MD attribute values.
Router(config)# end
Router#
000036: *Oct 20 16:26:04.570: %SYS-5-CONFIG_I: Configured from console by console
Router# show logging

Syslog logging: enabled (11 messages dropped, 0 messages rate-limited,
    0 flushes, 0 overruns, xml disabled, filtering disabled)
No Active Message Discriminator.
Inactive Message Discriminator:
ATTFLTR1  severity group includes 1,2,5
    rate-limit not to exceed 100 messages per second
Console logging: level debugging, 25 messages logged, xml disabled, filtering disabled
Monitor logging: level debugging, 0 messages logged, xml disabled, filtering disabled
Buffer logging:  level debugging, 25 messages logged, xml disabled, filtering disabled
Logging Exception size (8192 bytes)
Count and timestamp logging messages: disabled
No active filter modules.
Trap logging: level debugging, 28 message lines logged
Logging to 172.25.126.15  (udp port 1300,  audit disabled, authentication disabled,
encryption disabled, link up),
    28 message lines logged,
    0 message lines rate-limited,
    0 message lines dropped-by-MD,
    xml disabled, sequence number disabled
    filtering disabled
Logging to 172.25.126.15  (tcp port 1307,  audit disabled, authentication disabled,
encryption disabled, link up),
    28 message lines logged,
    0 message lines rate-limited,
    0 message lines dropped-by-MD,
    xml disabled, sequence number disabled, filtering disabled
Logging to 172.20.1.1  (udp port 514,  audit disabled,
    authentication disabled, encryption disabled, link up),
    28 message lines logged,
    0 message lines rate-limited,
    0 message lines dropped-by-MD,
    xml disabled, sequence number disabled
    filtering disabled
Log Buffer (1000000 bytes):
```

The table below describes the significant fields shown in the output for the two preceding examples.

Table 7: show logging Field Descriptions

Field	Description
Syslog logging:	Shows the general state of system logging (enabled or disabled), the status of logged messages (number of messages dropped, rate-limited, or flushed), and whether XML formatting or ESM filtering is enabled.
No Active Message Discriminator	Indicates that a message discriminator is not being used.
Inactive Message Discriminator:	Identifies a configured message discriminator that has not been invoked.
Console logging:	Logging to the console port. Shows “disabled” or, if enabled, the severity level limit, number of messages logged, and whether XML formatting or ESM filtering is enabled. Corresponds to the configuration of the **logging console**, **logging console filtered**, or **logging console xml** command.
Monitor logging:	Logging to the monitor (all TTY lines). Shows “disabled” or, if enabled, the severity level limit, number of messages logged, and whether XML formatting or ESM filtering is enabled. Corresponds to the configuration of the **logging monitor**, **logging monitor filtered** or **logging monitor xml** command.
Buffer logging:	Logging to the standard syslog buffer. Shows “disabled” or, if enabled, the severity level limit, number of messages logged, and whether XML formatting or ESM filtering is enabled. Corresponds to the configuration of the **logging buffered**, **logging buffered filtered**, or **logging buffered xml** command.
Trap logging:	Logging to a remote host (syslog collector). Shows “disabled” or, if enabled, the severity level limit, number of messages logged, and whether XML formatting or ESM filtering is enabled. (The word “trap” means a trigger in the system software for sending error messages to a remote host.) Corresponds to the configuration of the **logging host** command. The severity level limit is set using the **logging trap** command.
SNMP logging	Displays whether SNMP logging is enabled, the number of messages logged, and the retransmission interval. If not shown on your platform, use the **show logging history** command.
Logging Exception size (8192 bytes)	Corresponds to the configuration of the **logging exception** command.
Count and timestamp logging messages:	Corresponds to the configuration of the **logging count** command.

Field	Description
No active filter modules.	Appears if no syslog filter modules are configured with the **logging filter** command. Syslog filter modules are Tcl script files used when the Embedded Syslog Manager (ESM) is enabled. ESM is enabled when any of the **filtered** keywords are used in the logging commands. If configured, the URL and filename of configured syslog filter modules will appear at this position in the output. Syslog filter modules are executed in the order in which they appear here.
Log Buffer (8192 bytes):	The value in parentheses corresponds to the configuration of the **logging buffered** *buffer-size* command. If no messages are currently in the buffer, the output ends with this line. If messages are stored in the syslog buffer, they appear after this line.

The following example shows that syslog messages from the system buffer are included, with time stamps. In this example, the software image does not support XML formatting or ESM filtering of syslog messages.

```
Router# show logging

Syslog logging:enabled (2 messages dropped, 0 flushes, 0 overruns)
    Console logging:disabled
    Monitor logging:level debugging, 0 messages logged
    Buffer logging:level debugging, 4104 messages logged
    Trap logging:level debugging, 4119 message lines logged
        Logging to 192.168.111.14, 4119 message lines logged
Log Buffer (262144 bytes):
Jul 11 12:17:49 EDT:%BGP-4-MAXPFX:No. of prefix received from 209.165.200.225
(afi 0) reaches 24, max 24
! THE FOLLOWING LINE IS A DEBUG MESSAGE FROM NTP.
! NOTE THAT IT IS NOT PRECEEDED BY THE % SYMBOL.
Jul 11 12:17:48 EDT: NTP: Maxslew = 213866
Jul 11 15:15:41 EDT:%SYS-5-CONFIG:Configured from
tftp://host.com/addc5505-rsm.nyiix
.Jul 11 15:30:28 EDT:%BGP-5-ADJCHANGE:neighbor 209.165.200.226 Up
.Jul 11 15:31:34 EDT:%BGP-3-MAXPFXEXCEED:No. of prefix received from
209.165.200.226 (afi 0):16444 exceed limit 375
.Jul 11 15:31:34 EDT:%BGP-5-ADJCHANGE:neighbor 209.165.200.226 Down BGP
Notification sent
.Jul 11 15:31:34 EDT:%BGP-3-NOTIFICATION:sent to neighbor 209.165.200.226 3/1
(update malformed) 0 bytes
 .
 .
 .
```

The software clock keeps an “authoritative” flag that indicates whether the time is authoritative (believed to be accurate). If the software clock has been set by a timing source (for example, via Network Time Protocol (NTP), the flag is set. If the time is not authoritative, it will be used only for display purposes. Until the clock is authoritative and the “authoritative” flag is set, the flag prevents peers from synchronizing to the software clock.

The table below describes the symbols that precede the time stamp.

Table 8: Time-Stamping Symbols for Syslog Messages

Symbol	Description	Example
*	Time is not authoritative: the software clock is not in sync or has never been set.	*15:29:03.158 UTC Tue Feb 25 2003:
(blank)	Time is authoritative: the software clock is in sync or has just been set manually.	15:29:03.158 UTC Tue Feb 25 2003:
.	Time is authoritative, but NTP is not synchronized: the software clock was in sync, but has since lost contact with all configured NTP servers.	.15:29:03.158 UTC Tue Feb 25 2003:

The following is sample output from the **show logging summary** command for a Cisco 12012 router. A number in the column indicates that the syslog contains that many messages for the line card. For example, the line card in slot 9 has 1 error message, 4 warning messages, and 47 notification messages.

Note For similar log counting on other platforms, use the **show logging count** command.

```
Router# show logging summary
+-----+-------+-------+-------+-------+-------+-------+-------+-------+
 SLOT | EMERG | ALERT | CRIT  | ERROR |WARNING| NOTICE| INFO  | DEBUG |
+-----+-------+-------+-------+-------+-------+-------+-------+-------+
|* 0* |     . |     . |     . |     . |     . |     . |     . |     . |
|  1  |       |       |       |       |       |       |       |       |
|  2  |       |       |       |     1 |     4 |    45 |       |       |
|  3  |       |       |       |       |       |       |       |       |
|  4  |       |       |       |     5 |     4 |    54 |       |       |
|  5  |       |       |       |       |       |       |       |       |
|  6  |       |       |       |       |       |       |       |       |
|  7  |       |       |       |    17 |     4 |    48 |       |       |
|  8  |       |       |       |       |       |       |       |       |
|  9  |       |       |       |     1 |     4 |    47 |       |       |
| 10  |       |       |       |       |       |       |       |       |
| 11  |       |       |       |    12 |     4 |    65 |       |       |
+-----+-------+-------+-------+-------+-------+-------+-------+-------+
Router#
```

The table below describes the logging level fields shown in the display.

Table 9: show logging summary Field Descriptions

Field	Description
SLOT	Indicates the slot number of the line card. An asterisk next to the slot number indicates the GRP card whose error message counts are not displayed. For information on the GRP card, use the **show logging** command.
EMERG	Indicates that the system is unusable.
ALERT	Indicates that immediate action is needed.
CRIT	Indicates a critical condition.

Field	Description
ERROR	Indicates an error condition.
WARNING	Indicates a warning condition.
NOTICE	Indicates a normal but significant condition.
INFO	Indicates an informational message only.
DEBUG	Indicates a debugging message.

Related Commands

Command	Description
clear logging	Clears messages from the logging buffer.
logging buffered	Enables system message logging to a local buffer.
logging count	Enables the error log count capability.
logging history size	Changes the number of syslog messages stored in the history table of the router.
logging linecard	Logs messages to an internal buffer on a line card and limits the logging messages displayed on terminal lines other than the console line to messages with a level at or above level.
service timestamps	Configures the system to time-stamp debugging or logging messages.
show logging count	Displays a summary of system error messages (syslog messages) by facility and severity.
show logging xml	Displays the state of system logging and the contents of the XML-specific logging buffer.

show logging count

To display a summary of the number of times certain system error messages are occuring, use the **show logging** command in privileged EXEC mode.

show logging count

Syntax Description

This command has no arguements or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(8)T	This command was introduced.

Usage Guidelines

To enable the error log count capability (syslog counting feature), use the **logging count** command in global configuration mode.

This feature works independently of the various settings of the other logging commands (such as [**no**] **logging on**, [**no**] **logging buffered**, and so on). In other words, turning off logging by other means does not stop the counting and timestamping from occuring.

This command displays information such as the number of times a particular system error message occurs and the time stamp of the last occurrence of the specified message. System error messages are grouped into logical units called “Facilities” based on Cisco IOS software components.

To determine if system error message counting is enabled, use the **show logging** command.

The **service timestamps** command configuration determines the timestamp format (shown in the “Last Time” column) of **show logging count** command output. There is not quite enough space for all options of the possible options (datetime, milliseconds, and timezone) of the **service timestamps datetime**command to be displayed at the same time. As a result, if **msec** is selected, **timezone** will not be displayed. If **show-timezone** is selected but not **msec**, then the time zone will be displayed.

Occasionally, the length of the message name plus the facility name contains too many characters to be printed on one line. The CLI attempts to keep the name and facility name on one line but, if necessary, the line will be wrapped, so that the first line contains the facility name and the second line contains the message name and the rest of the columns.

Examples

The following example shows the number of times syslog messages have occurred and the most recent time that each error message occurred. In this example, the **show logging** command is used to determine if the syslog counting feature is enabled:

```
Router# show logging | include count
Count and timestamp logging messages: enabled
Router# show logging count
Facility       Message Name                     Sev  Occur  Last Time
==============================================================================
SYS            BOOTTIME                           6    1   00:00:12
SYS            RESTART                            5    1   00:00:11
SYS            CONFIG_I                           5    1   00:00:05
-------------  ------------------------------  ------------------------------
SYS TOTAL                                            3
LINEPROTO      UPDOWN                             5   13   00:00:19
-------------  ------------------------------  ------------------------------
LINEPROTO TOTAL                                     13
LINK           UPDOWN                             3    1   00:00:18
LINK           CHANGED                            5   12   00:00:09
-------------  ------------------------------  ------------------------------
LINK TOTAL                                          13
SNMP           COLDSTART                          5    1   00:00:11
-------------  ------------------------------  ------------------------------
SNMP TOTAL                                           1
```

The table below describes the significant fields shown in the display.

Table 10: show logging count Field Descriptions

Field	Description
Facility	The facility, such as syslog, from which these error messages are occurring.
Message Name	The name of this message.

Field	Description
Sev	The severity level of this message.
Occur	How many times this message has occurred.
Last Time	The last (most recent) time this message occurred. Timestamping is by default based on the system uptime (for example "3w1d" indicates 3 weeks and 1 day from the last system reboot.)
Sys Total / Lineproto Total / Link Total / SNMP Total	Total number of error messages that have occurred for the specified Facility.

In the following example, the user is interested only in the totals:

```
Router# show logging count | include total
SYS TOTAL                                              3
LINEPROTO TOTAL                                       13
LINK TOTAL                                            13
SNMP TOTAL                                             1
```

Related Commands

Command	Description
clear logging	Clears messages from the logging buffer.
logging count	Enables the system error message log count capability.
service timestamps	Configures the system to time-stamp debugging or logging messages.
show logging	Displays general information about the state of system logging.

show logging history

To display information about the state of the syslog history table, use the **show logging history** command in privileged EXEC mode.

show logging history

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

This command displays information about the syslog history table, such as the table size, the status of messages, and text of messages stored in the table. Messages stored in the table are governed by the **logging history** global configuration command.

Examples

The following example shows sample output from the **show logging history** command. In this example, notifications of severity level 5 (notifications) through severity level 0 (emergencies) are configured to be written to the logging history table.

```
Router# show logging history
Syslog History Table: 1 maximum table entries,
saving level notifications or higher
0 messages ignored, 0 dropped, 15 table entries flushed,
SNMP notifications not enabled
  entry number 16: SYS-5-CONFIG_I
  Configured from console by console
  timestamp: 1110
Router#
```

The table below describes the significant fields shown in the output.

Table 11: show logging history Field Descriptions

Field	Description
maximum table entry	Number of messages that can be stored in the history table. Set with the **logging history size** command.
saving level notifications <x> or higher	Level of messages that are stored in the history table and sent to the SNMP server (if SNMP notification is enabled). The severity level can be configured with the **logging history** command.
messages ignored	Number of messages not stored in the history table because the severity level is greater than that specified with the **logging history** command.
dropped	Number of messages that could not be processed due to lack of system resources. Dropped messages do not appear in the history table and are not sent to the SNMP server.
table entries flushed	Number of messages that have been removed from the history table to make room for newer messages.
SNMP notifications	Whether syslog traps of the appropriate level are sent to the SNMP server. The sending of syslog traps are enabled or disabled through the **snmp-server enable traps syslog** command.
entry number:	Number of the message entry in the history table. In the example above, the message "SYS-5-CONFIG_I Configured from console by console" indicates a syslog message consisting of the facility name (SYS), which indicates where the message came from, the severity level (5) of the message, the message name (CONFIG_I), and the message text.
timestamp	Time, based on the up time of the router, that the message was generated.

Related Commands

Command	Description
clear logging	Clears messages from the logging buffer.

Command	Description
logging history	Limits syslog messages sent to the router's history table to a specified severity level.
logging history size	Changes the number of syslog messages that can be stored in the history table.
logging linecard	Logs messages to an internal buffer on a line card. This command limits the logging messages displayed on terminal lines other than the console line to messages with a level at or above level.
snmp-server enable traps	The [**no**] **snmp-server enable traps syslog** form of this command controls (enables or disables) the sending of system-logging messages to a network management station.

show ip ports all

To display all the open ports on a device, use the **show ip ports all** in user EXEC or privileged EXEC mode.

show ip ports all

Syntax Description

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)EZ	This command was introduced on the Catalyst 3750-X and Catalyst 3560-X Switches.
Cisco IOS XE Everest 16.5.1	This command was implemented on Cisco ASR1000 Aggregation Series Routers, Cisco 4000 Series Integrated Services Routers and Cisco Cloud Services Router 1000V Series.

Usage Guidelines

This command provides a list of all open TCP/IP ports on the system including the ports opened using Cisco networking stack.

Examples

The following is sample output from the **show ip ports all** command:

```
Device# show ip ports all

Proto Local Address              Foreign Address            State       PID/Program Name
TCB   Local Address              Foreign Address            (state)

tcp   *:4786                     *:*                        LISTEN      221/[IOS]SMI IBC
 server process
udp   *:2228                     0.0.0.0:0                              297/[IOS]L2TRACE
```

```
 SERVER
------------------ TCP/UDP ports used by various Linux processes ----------------------

Proto   Local Address           Foreign Address       STATE           PID/ Process name

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

 tcp    127.0.0.1:7022      0.0.0.0:*           LISTEN 9519      /sshd

 udp    0.0.0.0:1812        0.0.0.0:*                  25668     /smd

 udp    0.0.0.0:1813        0.0.0.0:*                  25668     /smd

udp6    :::1812             :::*                       25668     /smd

udp6    :::1813             :::*                       25668     /smd

 tcp    127.0.0.1:7022      0.0.0.0:*           LISTEN 9519      /sshd

After reconfiguring ip http server.

Device# show ip ports all

Proto Local Address               Foreign Address             State       PID/Program Name

TCB   Local Address               Foreign Address             (state)

tcp   :::443                      :::*                        LISTEN      286/[IOS]HTTP
CORE

tcp   *:443                       *:*                         LISTEN      286/[IOS]HTTP
CORE

tcp   :::80                       :::*                        LISTEN      286/[IOS]HTTP
CORE

tcp   *:80                        *:*                         LISTEN      286/[IOS]HTTP
CORE

tcp   *:4786                      *:*                         LISTEN      221/[IOS]SMI IBC
 server process

udp   *:2228                      0.0.0.0:0                               297/[IOS]L2TRACE
 SERVER
```

The table below describes the significant fields shown in the display

Table 12: Field Descriptions of show ip ports all

Field	Description
Protocol	Transport protocol used.
Local Address.	Device IP Address.
Foreign Address	Remote or peer address.
State	State of the connection. It can be listen, established or connected.
PID/Program Name	Process ID or name

Related Commands

Command	Description
show tcp brief all	Displays information about TCP connection endpoints.
show ip sockets	Displays IP sockets information.

show logging system

To display the System Event Archive (SEA) logs, use the **show logging system** command in user EXEC mode or privileged EXEC mode.

show logging system [{**disk** [*file-location*] | **last** [*num-of-last-log-msgs*]}]

Syntax Description

disk	(Optional) Displays SEA log disk, where the logs will be stored.
disk *file-location*	(Optional) Displays SEA logs from the specified file location. The **disk** keyword when used along with *file-location* argument displays SEA logs from the specified file location.
num-of-last-log-msgs	(Optional) Displays the specified number of log messages.

Command Default

This command has no default settings.

Command Modes

User EXEC (>) Privileged EXEC (#)

Command History

Release	Modification
12.2(33)SXH	This command was introduced.
12.2(33)SCC	This command was introduced for the Cisco uBR10012 Router in the Cisco IOS Software Release 12.2(33)SCC.

Usage Guidelines

The **show logging system**command displays the latest messages first.

Examples

The following example shows a sample output of the show logging system command that displays the specified number of latest system log messages:

```
Router# show logging system
SEQ: MM/DD/YY HH:MM:SS MOD/SUB: SEV, COMP, MESSAGE
=====================================================
1: 01/24/07 15:38:40 6/-1 : MAJ, GOLD, syndiagSyncPinnacle failed in slot 6
2: 01/24/07 15:38:40 6/-1 : MAJ, GOLD, queryHyperionSynched[6]: Hyperion out of sync in
sw_mode 1
3: 01/24/07 15:38:40 6/-1 : MAJ, GOLD, queryHyperionSynched[6]: Hyperion out of sync in
sw_mode 1
4: 01/24/07 15:38:40 6/-1 : MAJ, GOLD, queryHyperionSynched[6]: Hyperion out of sync in
sw_mode 1
5: 01/24/07 15:38:40 6/-1 : MAJ, GOLD, queryHyperionSynched[6]: Hyperion out of sync in
sw_mode 1
6: 01/24/07 15:38:40 6/-1 : MAJ, GOLD, queryHyperionSynched[6]: Hyperion out of sync in
sw_mode 1
```

```
7: 01/24/07 15:38:39 6/-1 : MAJ, GOLD, queryHyperionSynched[6]: Hyperion out of sync in
sw_mode 1
```

The table below describes the significant fields shown in the display.

Table 13: show logging system Field Descriptions

Field	Description
MOD/SUB	Module or the submodule that generated the log message.
SEV	Severity level of the message.
COMP	Software component that has logged the message.

The following example shows a sample output of the show logging system command that displays SEA logs from the specified file location:

```
Router# show logging system disk disk0:my_log.dat
SEQ: MM/DD/YY HH:MM:SS MOD/SUB: SEV, COMP, MESSAGE
=====================================================
1: 02/01/95 00:35:51     2/3/-1: MAJ, GOLD, lc_ctrl_proc_obfl_info:test SEA log in
DFC:Diagnostic OBFL testing
2: 02/01/95 00:35:09     2/5/-1: MAJ, GOLD, diag_hit_sys_limit[3/2]: sp_netint_thr[0]
3: 02/01/95 00:35:09     2/5/-1: MAJ, GOLD, diag_hit_sys_limit[3/2]: SP[81%],Tx_rate[408],
 Rx_rate[0]
4: 02/01/95 00:35:08     2/5/-1: MAJ, GOLD, diag_hit_sys_limit[3/2]: sp_netint_thr[0]
5: 02/01/95 00:35:08     2/5/-1: MAJ, GOLD, diag_hit_sys_limit[3/2]: SP[82%],Tx_rate[453],
 Rx_rate[0]
6: 02/01/95 00:35:08     2/5/-1: MAJ, GOLD, test_c2cot_hm_ch0_test[3]: port 13, chnl 0,
Skipped Fabric Channel HM Test
7: 02/01/95 00:35:08     2/5/-1: MAJ, GOLD,
fabric_hm_inband_loopback_test[3/13]:diag_hit_sys_limit!test skipped.
8: 02/01/95 00:35:08     2/5/-1: MAJ, GOLD, diag_hit_sys_limit[3/13]: sp_netint_thr[0]
9: 02/01/95 00:35:08    2/5/-1: MAJ, GOLD, diag_hit_sys_limit[3/13]: SP[83%], Tx_rate[453],
 Rx_rate[0]
```

Cisco uBR10012 Universal Broadband Router

The following example shows a sample output of the **show logging system** command on the Cisco uBR10012 Router:

```
Router# show logging system

SEQ: MM/DD/YY HH:MM:SS MOD/SUB: SEV, COMP,    MESSAGE
=====================================================
1: 05/06/09 04:10:11     6/0: NON, SEATEST, "Test disk1":"
```

The following command is used to identify the disk on PRE currently being used to store the sea_log.dat file. The following example shows a sample output of the **show logging system disk** command executed on the Cisco uBR10012 router:

```
Router# show logging system
 disk
SEA log disk: disk1:
The following command is used to view the specified number of log messages stored in the
sea_log.dat file. The following example shows a sample output of the show logging system
```

```
 last 10
command on the Cisco uBR10012 router:
Router# show logging system
 last 10
SEQ: MM/DD/YY HH:MM:SS MOD/SUB: SEV, COMP,    MESSAGE
=====================================================
1: 05/06/09 04:47:48 5/0: NON, SEATEST, "Second Message"
2: 05/06/09 04:47:31 6/0: NON, SEATEST, "First Message"
```

Related Commands

clear logging system	Clears the event records stored in the SEA.
copy logging system	Copies the archived system events to another location.
logging system	Enables or disables the SEA logging system.

show logging xml

To display the state of system message logging in an XML format, and to display the contents of the XML syslog buffer, use the **show logging xml** command in privileged EXEC mode.

show logging xml

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(15)T	This command was introduced.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRE	This command was integrated into Cisco IOS Release 12.2(33)SRE.

Usage Guidelines

This command displays the same syslog state information as the standard **show logging** command, but displays the information in XML format. This command also displays the content of the XML syslog buffer (if XML-formatted buffer logging is enabled).

Examples

The following example compares the output of the standard **show logging** command with the output of the **show logging xml** command so that you can see how the standard information is formatted in XML.

```
Router# show logging
Syslog logging: enabled (10 messages dropped, 6 messages rate-limited, 0 flushes, 0 overruns,
 xml enabled)
    Console logging: level debugging, 28 messages logged, xml enabled
    Monitor logging: level debugging, 0 messages logged, xml enabled
    Buffer logging: level debugging, 2 messages logged, xml enabled (2 messages logged)
    Logging Exception size (8192 bytes)
    Count and timestamp logging messages: disabled
    Trap logging: level informational, 35 message lines logged
        Logging to 10.2.3.4, 1 message lines logged, xml disabled
```

```
        Logging to 192.168.2.1, 1 message lines logged, xml enabled

Log Buffer (8192 bytes):
00:04:20: %SYS-5-CONFIG_I: Configured from console by console
00:04:41: %SYS-5-CONFIG_I: Configured from console by console
Router# show logging xml
<syslog-logging status="enabled" msg-dropped="10" msg-rate-limited="6" flushes="0"
overruns="0"><xml>enabled</xml></syslog-logging>
    <console-logging level="debugging"
messages-logged="28"><xml>enabled</xml></console-logging>
    <monitor-logging level="debugging"
messages-logged="0"><xml>enabled</xml></monitor-logging>
    <buffer-logging level="debugging" messages-logged="2"><xml
messages-logged="2">enabled</xml></buffer-logging>
    <logging-exception size="8192 bytes"></logging-exception>
    <count-and-timestamp-logging status="disabled"></count-and-timestamp-logging>
    <trap-logging level="informational" messages-lines-logged="35"></trap-logging>
        <logging-to><dest id="0" ipaddr="10.2.3.4"
message-lines-logged="1"><xml>disabled</xml><dest></logging-to>
        <logging-to><dest id="1" ipaddr="192.168.2.1"
message-lines-logged="1"><xml>enabled</xml><dest></logging-to>

<log-xml-buffer size="44444 bytes">></log-xml-buffer>
<ios-log-msg><facility>SYS</facility><severity>5</severity><msg-id>CONFIG_I</msg-id><time>00:04:20</time><args><arg
 id="0">console</arg><arg id="1">console</arg></args></ios-log-msg>
<ios-log-msg><facility>SYS</facility><severity>5</severity><msg-id>CONFIG_I</msg-id><time>00:04:41</time><args><arg
 id="0">console</arg><arg id="1">console</arg></args></ios-log-msg>
Router#
```

The table below describes the significant fields shown in the displays.

Table 14: show logging and show logging xml Field Descriptions

Field	Description	XML Tag
Syslog logging	The global state of system message logging (syslog); “enabled” or “disabled.”	syslog-logging
Console logging	State of logging to console connections.	console-logging
Monitor logging	State of logging to monitor (TTY and Telnet) connections.	monitor-logging
Buffer logging	State of logging to the local system logging buffer.	buffer-logging
Count and timestamp logging messages:	Indicates whether the logging count feature is enabled. Corresponds to the **logging count** command.	count-and-timestamp-logging
Trap logging	State of logging to a remote host.	trap-logging

Related Commands

Command	Description
show logging	Displays the contents of the standard syslog buffer.
show logging count	Displays counts of each system error message.

Command	Description
show logging history	Displays the contents of the SNMP syslog history table.

show memory

To display statistics about memory when Cisco IOS software, Cisco IOS XE or Software Modularity images are running, use the **show memory** command in user EXEC or privileged EXEC mode.

Cisco IOS software
show memory [*memory-type*] [**free**] [**overflow**] [**summary**] [**poisoning**]

Cisco IOS XE or Software Modularity
show memory

Syntax Description

memory-type	(Optional) Memory type to display (**processor**, **multibus**, **io**, or **sram**). If *memory-type* is not specified, statistics for all memory types present are displayed.
free	(Optional) Displays free memory statistics.
overflow	(Optional) Displays details about memory block header corruption corrections when the **exception memory ignore overflow** global configuration command is configured.
summary	(Optional) Displays a summary of memory usage including the size and number of blocks allocated for each address of the system call that allocated the block.
poisoning	(Optional) Displays memory poisoning details, including the following: • Alloc PID • Alloc Check • Alloc PC • Alloc Name • Corrupt Ptr • Corrupt Val • TotalBytes • MarkedBytes • TIME

Command Modes

User EXEC (>) Privileged EXEC (#)

Command History

Release	Modification
10.0	This command was introduced.

Release	Modification
12.3(7)T	This command was enhanced with the **overflow** keyword to display details about memory block header corruption corrections.
12.2(25)S	The command output was updated to display information about transient memory pools.
12.3(14)T	The command output was updated to display information about transient memory pools.
12.2(27)SBC	This command was integrated into Cisco IOS Release 12.2(27)SBC.
12.2(18)SXF4	This command was implemented in Cisco IOS Software Modularity images.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(20)T	The **poisoning** keyword was added.
Cisco IOS XE Release 3.1.0.SG	The **show memory** stand-alone command was introduced on the Cisco Catalyst 4500e Serfies Switches. The command functions as shown in the Cisco IOS XE or Software Modularity examples.

Usage Guidelines

Cisco IOS Software

The **show memory** command displays information about memory available after the system image decompresses and loads.

Cisco IOS XE or Software Modularity

Use the **show memory** command when a Cisco IOS XE or Software Modularity image is running to display a summary of system-wide memory utilization. To display details about POSIX and Cisco IOS style system memory information when Software Modularity images are running, use the **show memory detailed** command.

Examples

Example output varies between Cisco IOS software images and Cisco IOS Software Modularity software images. To view the appropriate output, see the following sections:

- Cisco IOS Software
- Cisco IOS XE
- Cisco IOS Software Modularity

Cisco IOS Software

The following is sample output from the **show memory** command:

```
Router# show memory
               Head   Total(b)    Used(b)    Free(b)  Lowest(b) Largest(b)
Processor    B0EE38    5181896    2210036    2971860    2692456    2845368
          Processor memory
Address   Bytes Prev.    Next     Ref  PrevF   NextF   Alloc PC  What
B0EE38     1056 0        B0F280     1                  18F132    List Elements
B0F280     2656 B0EE38   B0FD08     1                  18F132    List Headers
```

```
B0FD08     2520 B0F280   B10708     1                  141384    TTY data
B10708     2000 B0FD08   B10F00     1                  14353C    TTY Input Buf
B10F00      512 B10708   B11128     1                  14356C    TTY Output Buf
B11128     2000 B10F00   B11920     1                  1A110E    Interrupt Stack
B11920       44 B11128   B11974     1                  970DE8    *Init*
B11974     1056 B11920   B11DBC     1                  18F132    messages
B11DBC       84 B11974   B11E38     1                  19ABCE    Watched Boolean
B11E38       84 B11DBC   B11EB4     1                  19ABCE    Watched Boolean
B11EB4       84 B11E38   B11F30     1                  19ABCE    Watched Boolean
B11F30       84 B11EB4   B11FAC     1                  19ABCE    Watched Boolean
```

The following is sample output from the **show memory free** command:

```
Router# show memory free
               Head   Total(b)    Used(b)    Free(b)  Lowest(b) Largest(b)
Processor      B0EE38    5181896    2210076    2971820    2692456    2845368
          Processor memory
Address   Bytes Prev.    Next     Ref  PrevF   NextF   Alloc PC  What
             24    Free list 1
CEB844       32  CEB7A4 CEB88C      0  0       0       96B894    SSE Manager
             52    Free list 2
             72    Free list 3
             76    Free list 4
             80    Free list 5
D35ED4       80 D35E30   D35F4C     0  0       D27AE8  96B894    SSE Manager
D27AE8       80 D27A48   D27B60     0  D35ED4  0       22585E    SSE Manager
             88    Free list 6
            100    Free list 7
D0A8F4      100 D0A8B0   D0A980     0  0       0       2258DA    SSE Manager
            104    Free list 8
B59EF0      108 B59E8C   B59F84     0  0       0       2258DA    (fragment)
```

The output of the **show memory free** command contains the same types of information as the **show memory** output, except that only free memory is displayed, and the information is ordered by free list.

The first section of the display includes summary statistics about the activities of the system memory allocator. The table below describes the significant fields shown in the first section of the display.

Table 15: show memory Field Descriptions--First Section

Field	Description
Head	Hexadecimal address of the head of the memory allocation chain.
Total(b)	Sum of used bytes plus free bytes.
Used(b)	Amount of memory in use.
Free(b)	Amount of memory not in use.
Lowest(b)	Smallest amount of free memory since last boot.
Largest(b)	Size of largest available free block.

The second section of the display is a block-by-block listing of memory use. The table below describes the significant fields shown in the second section of the display.

Table 16: Characteristics of Each Block of Memory--Second Section

Field	Description
Address	Hexadecimal address of block.
Bytes	Size of block (in bytes).
Prev.	Address of previous block (should match the address on previous line).
Next	Address of next block (should match the address on next line).
Ref	Reference count for that memory block, indicating how many different processes are using that block of memory.
PrevF	Address of previous free block (if free).
NextF	Address of next free block (if free).
Alloc PC	Address of the system call that allocated the block.
What	Name of process that owns the block, or "(fragment)" if the block is a fragment, or "(coalesced)" if the block was coalesced from adjacent free blocks.

The **show memory io** command displays the free I/O memory blocks. On the Cisco 4000 router, this command quickly shows how much unused I/O memory is available.

The following is sample output from the **show memory io** command:

```
Router# show memory io
Address   Bytes Prev.   Next     Ref  PrevF   NextF   Alloc PC  What
6132DA0   59264 6132664 6141520  0    0      600DDEC  3FCF0     *Packet Buffer*
600DDEC     500 600DA4C 600DFE0  0   6132DA0 600FE68  0
600FE68     376 600FAC8 600FFE0  0   600DDEC 6011D54  0
6011D54     652 60119B4 6011FEO  0   600FE68 6013D54  0
614FCA0     832 614F564 614FFE0  0   601FD54 6177640  0
6177640 2657056 6172E90 0        0   614FCA0 0        0
Total: 2723244
```

The following sample output displays details of a memory block overflow correction when the **exception memory ignore overflow** global configuration command is configured:

```
Router# show memory overflow
Count   Buffer Count     Last corrected      Crashinfo files
1       1                00:11:17            slot0:crashinfo_20030620-075755
Traceback   607D526C 608731A0 607172F8 607288E0 607A5688 607A566C
```

The report includes the amount of time since the last correction was made and the name of the file that logged the memory block overflow details.

The **show memory sram** command displays the free SRAM memory blocks. For the Cisco 4000 router, this command supports the high-speed static RAM memory pool to make it easier for you to debug or diagnose problems with allocation or freeing of such memory.

The following is sample output from the **show memory sram** command:

```
Router# show memory sram
Address   Bytes Prev.   Next     Ref  PrevF   NextF   Alloc PC  What
```

```
7AE0      38178 72F0    0        0    0       0       0
Total     38178
```

The following sample output from the **show memory** command used on the Cisco 4000 router includes information about SRAM memory and I/O memory:

```
Router# show memory
               Head   Total(b)    Used(b)    Free(b)  Lowest(b) Largest(b)
Processor    49C724   28719324    1510864   27208460   26511644   15513908
      I/O   6000000    4194304    1297088    2897216    2869248    2896812
     SRAM      1000      65536      63400       2136       2136       2136
Address   Bytes Prev.   Next     Ref  PrevF   NextF   Alloc PC  What
1000       2032 0       17F0       1                   3E73E     *Init*
17F0       2032 1000    1FE0       1                   3E73E     *Init*
1FE0        544 17F0    2200       1                   3276A     *Init*
2200         52 1FE0    2234       1                   31D68     *Init*
2234         52 2200    2268       1                   31DAA     *Init*
2268         52 2234    229C       1                   31DF2     *Init*
72F0       2032 6E5C    7AE0       1                   3E73E     Init
7AE0      38178 72F0    0          0    0      0      0
```

The **show memory summary** command displays a summary of all memory pools and memory usage per Alloc PC (address of the system call that allocated the block).

The following is a partial sample output from the **show memory summary** command. This output shows the size, blocks, and bytes allocated. Bytes equal the size multiplied by the blocks. For a description of the other fields, see the tables above.

```
Router# show memory summary
Head   Total(b)    Used(b)    Free(b)  Lowest(b) Largest(b)
Processor    B0EE38    5181896    2210216    2971680    2692456    2845368
          Processor memory
Alloc PC        Size     Blocks      Bytes    What
0x2AB2           192          1        192    IDB: Serial Info
0x70EC            92          2        184    Init
0xC916           128         50       6400    RIF Cache
0x76ADE         4500          1       4500    XDI data
0x76E84         4464          1       4464    XDI data
0x76EAC          692          1        692    XDI data
0x77764          408          1        408    Init
0x77776          116          1        116    Init
0x777A2          408          1        408    Init
0x777B2          116          1        116    Init
0xA4600           24          3         72    List
0xD9B5C           52          1         52    SSE Manager
.
.
.
0x0                0       3413    2072576    Pool Summary
0x0                0         28    2971680    Pool Summary (Free Blocks)
0x0               40       3441     137640    Pool Summary (All Block Headers)
0x0                0       3413    2072576    Memory Summary
0x0                0         28    2971680    Memory Summary (Free Blocks)
```

Cisco IOS XE

The following is sample output from the **show memory** command when a Cisco IOS XE image is running.

```
Router# show memory
#show memory
System memory   : 1943928K total, 735007K used, 1208921K free, 153224K kernel reserved
Lowest(b)       : 641880064
                Total(K)     Used(K)      Free(K)
Process         1141112      514129       626984
Config          802816       220879       581937
```

The table below describes the significant fields shown in the display.

Table 17: show memory (Software Modularity Image) Field Descriptions

Field	Description
total	Total amount of memory on the device, in kilobytes.
used	Amount of memory in use, in kilobytes.
free	Amount of memory not in use, in kilobytes.
kernel reserved	Amount of memory reserved by the kernel, in kilobytes.
Process	Amount of memory used by processes.
Config	Amount of memory used by the configuration.

Cisco IOS Software Modularity

The following is sample output from the **show memory** command when a Cisco IOS Software Modularity image is running.

```
Router# show memory
System Memory: 262144K total, 116148K used, 145996K free 4000K kernel reserved
```

The table below describes the significant fields shown in the display.

Table 18: show memory (Software Modularity Image) Field Descriptions

Field	Description
total	Total amount of memory on the device, in kilobytes.
used	Amount of memory in use, in kilobytes.
free	Amount of memory not in use, in kilobytes.
kernel reserved	Amount of memory reserved by the kernel, in kilobytes.

Related Commands

Command	Description
exception memory ignore overflow	Configures the Cisco IOS software to correct corruptions in memory block headers and allow a router to continue its normal operation.

Command	Description
show memory detailed	Displays POSIX and Cisco IOS style system memory information.
show processes memory	Displays memory used per process.

show memory allocating-process

To display statistics on allocated memory with corresponding allocating processes, use the **show memory allocating-process** command in user EXEC or privileged EXEC mode.

show memory allocating-process [**totals**]

Syntax Description

totals	(Optional) Displays allocating memory totals.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.0	This command was introduced.

Usage Guidelines

The **show memory allocating-process**command displays information about memory available after the system image decompresses and loads.

Examples

The following is sample output from the **show memory allocating-process** command:

```
Router# show memory allocating-process
 Head  Total(b) Used(b) Free(b) Lowest(b)  Largest(b)
Processor  44E03560 186632636 26131896 160500740 160402052  153078204
     Fast  44DE3560 131072 58280 72792 72792 72764
          Processor memory
 Address  Bytes Prev.    Next     Ref    Alloc Proc    Alloc PC  What
6148EC40   1504 0        6148F24C   1  *Init*          602310FC  List Elements
6148F24C   3004 6148EC40 6148FE34   1  *Init*          60231128  List Headers
6148FE34   9000 6148F24C 61492188   1  *Init*          6023C634  Interrupt Stack
61492188     44 6148FE34 614921E0   1  *Init*          60C17FD8  *Init*
614921E0   9000 61492188 61494534   1  *Init*          6023C634  Interrupt Stack
61494534     44 614921E0 6149458C   1  *Init*          60C17FD8  *Init*
6149458C    220 61494534 61494694   1  *Init*          602450F4  *Init*
61494694   4024 6149458C 61495678   1  *Init*          601CBD64  TTY data
.
.
.
```

The table below describes the significant fields shown in the display.

Table 19: show memory allocating-process Field Descriptions

Field	Description
Head	Hexadecimal address of the head of the memory allocation chain.

Field	Description
Total(b)	Sum of used bytes plus free bytes.
Used(b)	Amount of memory in use in bytes.
Free(b)	Amount of memory not in use (in bytes).
Lowest(b)	Smallest amount of free memory since last boot (in bytes).
Largest(b)	Size of largest available free block (in bytes).
Address	Hexadecimal address of the block.
Bytes	Size of the block (in bytes).
Prev.	Address of the preceding block (should match the address on preceding row).
Next	Address of the following block (should match the address on following row).
Ref	Reference count for that memory block, indicating how many different processes are using that block of memory.
Alloc PC	Address of the system call that allocated the block.
What	Name of process that owns the block, or “(fragment)” if the block is a fragment, or “(coalesced)” if the block was coalesced from adjacent free blocks.

The following is sample output from the **show memory allocating-process totals**command:

```
Router# show memory allocating-process totals
                Head    Total(b)     Used(b)     Free(b)   Lowest(b)  Largest(b)
Processor   44E03560   186632636    26142524   160490112   160402052   153078204
     Fast   44DE3560      131072       58280       72792       72792       72764
Allocator PC Summary for: Processor
    PC          Total   Count  Name
0x4041AF8C    5710616    3189  *Packet Data*
0x4041AF40    2845480    3190  *Packet Header*
0x404DBA28    1694556     203  Process Stack
0x4066EA68    1074080      56  Init
0x404B5F68    1049296       9  pak subblock chunk
0x41DCF230     523924      47  TCL Chunks
0x404E2488     448920       6  MallocLite
0x4066EA8C     402304      56  Init
0x40033878     397108       1  Init
0x41273E24     320052       1  CEF: table event ring
0x404B510C     253152      24  TW Buckets
0x42248F0C     229428       1  Init
0x42248F28     229428       1  Init
0x42248F48     229428       1  Init
0x423FF210     218048       5  Dn48oC!M
0x421CB530     208144       1  epa crypto blk
0x417A07F0     196764       3  L2TP Hash Table
0x403AFF50     187836       3  Init
```

The table below describes the significant fields shown in the display.

Table 20: show memory allocating-process totals Field Descriptions

Field	Description
Head	Hexadecimal address of the head of the memory allocation chain.
Total(b)	Sum of used bytes plus free bytes.
Used(b)	Amount of memory in use (in bytes).
Free(b)	Amount of memory not in use (in bytes).
Lowest(b)	Smallest amount of free memory since last boot (in bytes).
Largest(b)	Size of the largest available free block in bytes.
PC	Program counter
Total	Total memory allocated by the process (in bytes).
Count	Number of allocations.
Name	Name of the allocating process.

Related Commands

Command	Description
show processes memory	Displays memory used per process.

show memory dead

To display statistics on memory allocated by processes that have terminated, use the show memory dead command in user EXEC or privileged EXEC mode.

show memory dead [**totals**]

Syntax Description

totals	(Optional) Displays memory totals for processes that have been terminated.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.0	This command was introduced.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

The **show memory dead**command displays information about processes that have been terminated. Terminated processes accounts for memory allocated under another process.

Examples

The following is sample output from the **show memory dead** command:

```
Router# show memory dead
    Head    Total(b)  Used(b)     Free(b)   Lowest(b)  Largest(b)
      I/O      600000    2097152    461024    1636128    1635224    1635960

          Processor memory

 Address  Bytes Prev.    Next     Ref  PrevF   NextF   Alloc PC  What
1D8310       60 1D82C8   1D8378     1                   3281FFE   Router Init
2CA964       36 2CA914   2CA9B4     1                   3281FFE   Router Init
2CAA04      112 2CA9B4   2CAAA0     1                   3A42144   OSPF Stub LSA RBTree
2CAAA0       68 2CAA04   2CAB10     1                   3A420D4   Router Init
2ED714       52 2ED668   2ED774     1                   3381C84   Router Init
2F12AC       44 2F124C   2F1304     1                   3A50234   Router Init
2F1304       24 2F12AC   2F1348     1                   3A420D4   Router Init
2F1348       68 2F1304   2F13B8     1                   3381C84   Router Init
300C28      340 300A14   300DA8     1                   3381B42   Router Init
```

The table below describes the significant fields shown in the display.

Table 21: show memory dead Field Descriptions

Field	Description
Head	Hexadecimal address of the head of the memory allocation chain.
Total(b)	Sum of used bytes plus free bytes.
Used(b)	Amount of memory in use.
Free(b)	Amount of memory not in use (in bytes).
Lowest(b)	Smallest amount of free memory since last boot (in bytes).
Largest(b)	Size of the largest available free block (in bytes).
Address	Hexadecimal address of the block (in bytes).
Bytes	Size of the block (in bytes).
Prev.	Address of the preceding block.
Next	Address of the following block.
Ref	Reference count for that memory block, indicating how many different processes are using that block of memory.
PrevF	Address of the preceding free block (if free).
NextF	Address of the following free block (if free).
Alloc PC	Address of the program counter that allocated the block.
What	Name of the process that owns the block, or “(fragment)” if the block is a fragment, or “(coalesced)” if the block was coalesced from adjacent free blocks.

show memory debug incremental

To display information about memory leaks after a starting time has been established, use the **show memory debug incremental** command in privileged EXEC mode.

show memory debug incremental {**allocations** | **leaks**[{**lowmem** | **summary**}] | **status**}

Syntax Description

allocations	Displays all memory blocks that were allocated after issuing the **set memory debug incremental starting-time** command.
leaks	Displays only memory that was leaked after issuing the **set memory debug incremental starting-time** command.
lowmem	(Optional) Forces the memory leak detector to work in low memory mode, making no memory allocations.
summary	(Optional) Reports summarized memory leaks based on allocator_pc and size of the memory block.
status	Displays all memory blocks that were allocated after issuing the **set memory debug incremental starting-time** command.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(7)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4T	The summary keyword was added.

Usage Guidelines

The **show memory debug incremental allocations** command displays all the memory blocks that were allocated after the **set memory debug incremental starting-time** command was entered. The displayed memory blocks are just memory allocations, they are not necessarily leaks.

The **show memory debug incremental leaks** command provides output similar to the **show memory debug leaks** command, except that it displays only memory that was leaked after the **set memory debug incremental starting-time** command was entered.

The **show memory debug incremental leaks lowmem** command forces memory leak detection to work in low memory mode. The amount of time taken for analysis is considerably greater than that of normal mode. The output for this command is similar to the **show memory debug leaks** command, except that it displays only memory that was leaked after the **set memory debug incremental starting-time** command was entered. You can use this command when you already know that normal mode memory leak detection will fail (perhaps by an unsuccessful previous attempt to invoke normal mode memory leak detection).

The **show memory debug incremental leaks summary** command displays a summarized report of the memory that was leaked after the **set memory debug incremental starting-time** command was entered, ordered by allocator process call address (Alloc_pc) and by memory block size.

The **show memory debug incremental status** command displays whether a starting point for incremental analysis has been set and the elapsed time since then.

Note All show memory debug commands must be used on customer networks only to diagnose the router for memory leaks when memory depletion is observed. These CLI's will have high CPU utilization and might result in time sensitive protocols to flap. These CLI's are recommended for customer use, only in the maintenance window when the router is not in a scaled condition.

Note All memory leak detection commands invoke normal mode memory leak detection, except when the low memory option is specifically invoked by use of the **lowmem** keyword. In normal mode, if memory leak detection determines that there is insufficient memory to proceed in normal mode, it will display an appropriate message and switch to low memory mode.

Examples

show memory debug incremental allocations Command Example

The following example shows output from the **show memory debug incremental**command when entered with the **allocations** keyword:

```
Router# show memory debug incremental allocations
Address    Size   Alloc_pc  PID  Name
62DA4E98      176 608CDC7C  44   CDP Protocol
62DA4F48       88 608CCCC8  44   CDP Protocol
62DA4FA0       88 606224A0  3    Exec
62DA4FF8       96 606224A0  3    Exec
635BF040       96 606224A0  3    Exec
63905E50      200 606A4DA4  69   Process Events
```

show memory debug incremental leaks summary Command Example

The following example shows output from the **show memory debug incremental** command when entered with the **leaks** and **summary** keywords:

```
Router# show memory debug incremental leaks summary
Adding blocks for GD...
                PCI memory
Alloc PC     Size     Blocks      Bytes    What
                I/O memory
Alloc PC     Size     Blocks      Bytes    What
                Processor memory
Alloc PC     Size          Blocks       Bytes      What
0x60874198  0000000052  0000000001  0000000052    Exec
0x60874198  0000000060  0000000001  0000000060    Exec
0x60874198  0000000100  0000000001  0000000100    Exec
0x60874228  0000000052  0000000004  0000000208    Exec
0x60874228  0000000060  0000000002  0000000120    Exec
0x60874228  0000000100  0000000004  0000000400   Exec
```

show memory debug incremental status Command Example

The following example shows output from the **show memory debug incremental** command entered with the **status** keyword:

```
Router# show memory debug incremental status
Incremental debugging is enabled
Time elapsed since start of incremental debugging: 00:00:10
```

Related Commands

Command	Description
set memory debug incremental starting-time	Sets the current time as the starting time for incremental analysis.
show memory debug leaks	Displays detected memory leaks.

show memory debug leaks

To display detected memory leaks, use the **show memory debug leaks** command in privileged EXEC mode. This command does not have a **no** form.

Cisco IOS software
show memory debug leaks [{**chunks** | **largest** | **lowmem** | **summary**}]

Cisco Catalyst 4500e Series Switches running IOS XE software
show memory debug leaks all [{**detailed** | **totals**}]

Syntax Description

all	Displays the information about leak block of the internal memory .
detailed	(Optional) Displays the detailed information about memory debug leak.
chunks	(Optional) Displays the memory leaks in chunks.
largest	(Optional) Displays the top ten leaking allocator_pcs based on size, and the total amount of memory they have leaked.
lowmem	(Optional) Forces the memory leak detector to work in low memory mode, making no memory allocations.
summary	(Optional) Reports summarized memory leaks based on allocator_pc and size of the memory block.
totals	(Optional) Displays summary report with the total number of each process.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS 12.3(8)T1	This command was introduced.

Release	Modification
Cisco IOS 12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
Cisco IOS 12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
Cisco IOS XE 3.1.0.SG	This command was introduced on the Cisco Catalyst 4500e Serfies Switches to display per-process memory leak amounts.
Cisco IOS 15.2(2)E	This command was integrated into Cisco IOS Release 15.2(2)E.

Usage Guidelines

If optional keywords are not specified, the **show memory debug leaks** command invokes normal mode memory leak detection and does not look for memory leaks in chunks.

The **show memory debug leaks chunks** command invokes normal mode memory leak detection and looks for leaks in chunks as well.

The **show memory debug leaks largest** command displays the top ten leaking allocator_pcs and the total amount of memory that they have leaked. Additionally, each time when this command is invoked, it remembers the report of the previous invocation and compares it with the report of the current invocation. If there are new entries in the current report, they are tagged as “inconclusive.” If the same entry appears in the report of the previous invocation and the report of the current invocation, the inconclusive tag is not added. It is beneficial to run memory leak detection more than once and to consider only the consistently reported leaks.

The **show memory debug leaks lowmem** command forces memory leak detection to work in low memory mode. The amount of time taken for analysis is considerably greater than that of normal mode. The output for this command is similar to the **show memory debug leaks** command. You can use this command when you know that the normal mode memory leak detection fails (perhaps by an unsuccessful previous attempt to invoke normal mode memory leak detection).

The **show memory debug leaks summary** command reports memory leaks based on allocator_pc and then on the size of the block.

The **show memory debug leaks all detailed** command provides the details of memory leaks for a particular process.

The **show memory debug leaks all totals** command provides the summary report with the total number of memory leaks of each running process.

Note All **show memory debug** commands must be used on customer networks only to diagnose the router for memory leaks when memory depletion is observed. These CLIs have high CPU utilization and might result in time sensitive protocols to flap. These CLIs are recommended for customer use, only in the maintenance window when the router is not in a scaled condition.

Note The command **show memory debug leak lowmem** is extremely CPU intensive and can result in CPUHOG/WATCHDOG crash. This command must be used only when the router has reached an unusable state due to memory exhaustion. Its use on high end platforms such as ISR and above can potentially crash the box. Use of this command outside of these limitations can cause a console hang of one hour in some cases. As an alternative, use the **show memory debug leak** command.

Cisco IOS Software

show memory debug leaks Command Example

The following example shows output from the **show memory debug leaks** command:

```
Device# show memory debug leaks
Adding blocks for GD...
                PCI memory
Address    Size   Alloc_pc  PID  Name
                I/O memory
Address    Size   Alloc_pc  PID  Name
                Processor memory
Address    Size   Alloc_pc  PID  Name
62DABD28       80 60616750  -2   Init
62DABD78       80 606167A0  -2   Init
62DCF240       88 605B7E70  -2   Init
62DCF298       96 605B7E98  -2   Init
62DCF2F8       88 605B7EB4  -2   Init
62DCF350       96 605B7EDC  -2   Init
63336C28      104 60C67D74  -2   Init
63370D58       96 60C656AC  -2   Init
633710A0      304 60C656AC  -2   Init
63B2BF68       96 60C659D4  -2   Init
63BA3FE0    32832 608D2848  104  Audit Process
63BB4020    32832 608D2FD8  104  Audit Process
```

The table below describes the significant fields shown in the display.

Table 22: show memory debug leaks Field Descriptions

Field	Description
Address	Hexadecimal address of the leaked block.
Size	Size of the leaked block (in bytes).
Alloc_pc	Address of the system call that allocated the block.
PID	The process identifier of the process that allocated the block.
Name	The name of the process that allocated the block.

show memory debug leaks chunks Command Example

The following example shows output from the **show memory debug leaks chunks** command:

```
Router# show memory debug leaks chunks
Adding blocks for GD...
                PCI memory
Address    Size   Alloc_pc  PID  Name
Chunk Elements:
Address  Size  Parent   Name
                I/O memory
Address    Size   Alloc_pc  PID  Name
```

```
Chunk Elements:
Address  Size  Parent   Name
                 Processor memory
Address    Size   Alloc_pc  PID  Name
62DABD28       80 60616750  -2   Init
62DABD78       80 606167A0  -2   Init
62DCF240       88 605B7E70  -2   Init
62DCF298       96 605B7E98  -2   Init
62DCF2F8       88 605B7EB4  -2   Init
62DCF350       96 605B7EDC  -2   Init
63336C28      104 60C67D74  -2   Init
63370D58       96 60C656AC  -2   Init
633710A0      304 60C656AC  -2   Init
63B2BF68       96 60C659D4  -2   Init
63BA3FE0    32832 608D2848  104  Audit Process
63BB4020    32832 608D2FD8  104  Audit Process
Chunk Elements:
Address  Size  Parent   Name
62D80DA8    16 62D7BFD0 (Managed Chunk )
62D80DB8    16 62D7BFD0 (Managed Chunk )
62D80DC8    16 62D7BFD0 (Managed Chunk )
62D80DD8    16 62D7BFD0 (Managed Chunk )
62D80DE8    16 62D7BFD0 (Managed Chunk )
62E8FD60   216 62E8F888 (IPC Message He)
```

The table below describes the significant fields shown in the display.

Note **show memory debug leaks chunks** command is a debug command and CPU intensive. Hence, the trace-backs is thrown on running the command as expected.

Table 23: show memory debug leaks chunks Field Descriptions

Field	Description
Address	Hexadecimal address of the leaked block.
Size	Size of the leaked block (in bytes).
Alloc_pc	Address of the system call that allocated the block.
PID	The process identifier of the process that allocated the block.
Name	The name of the process that allocated the block.
Size	(Chunk Elements) Size of the leaked element (bytes).
Parent	(Chunk Elements) Parent chunk of the leaked chunk.
Name	(Chunk Elements) The name of the leaked chunk.

show memory debug leaks largest Command Example

The following example shows output from the **show memory debug leaks largest** command:

```
Router# show memory debug leaks largest
```

```
Adding blocks for GD...
                PCI memory
Alloc_pc    total leak size
                I/O memory
Alloc_pc    total leak size
                Processor memory
Alloc_pc    total leak size
608D2848    32776     inconclusive
608D2FD8    32776     inconclusive
60C656AC    288       inconclusive
60C67D74    48        inconclusive
605B7E98    40        inconclusive
605B7EDC    40        inconclusive
60C659D4    40        inconclusive
605B7E70    32        inconclusive
605B7EB4    32        inconclusive
60616750    24        inconclusive
```

The following example shows output from the second invocation of the **show memory debug leaks largest** command:

```
Router# show memory debug leaks largest
Adding blocks for GD...
                PCI memory
Alloc_pc    total leak size
                I/O memory
Alloc_pc    total leak size
                Processor memory
Alloc_pc    total leak size
608D2848    32776
608D2FD8    32776
60C656AC    288
60C67D74    48
605B7E98    40
605B7EDC    40
60C659D4    40
605B7E70    32
605B7EB4    32
60616750    24
```

show memory debug leaks summary Command Example

The following example shows output from the **show memory debug leaks summary** command:

```
Router# show memory debug leaks summary
Adding blocks for GD...
                PCI memory
Alloc PC        Size     Blocks      Bytes    What
                I/O memory
Alloc PC        Size     Blocks      Bytes    What
                Processor memory
Alloc PC        Size     Blocks      Bytes    What
0x605B7E70 0000000032 0000000001 0000000032    Init
0x605B7E98 0000000040 0000000001 0000000040    Init
0x605B7EB4 0000000032 0000000001 0000000032    Init
0x605B7EDC 0000000040 0000000001 0000000040    Init
0x60616750 0000000024 0000000001 0000000024    Init
0x606167A0 0000000024 0000000001 0000000024    Init
0x608D2848 0000032776 0000000001 0000032776    Audit Process
0x608D2FD8 0000032776 0000000001 0000032776    Audit Process
0x60C656AC 0000000040 0000000001 0000000040    Init
```

```
0x60C656AC 0000000248 0000000001 0000000248    Init
0x60C659D4 0000000040 0000000001 0000000040    Init
0x60C67D74 0000000048 0000000001 0000000048    Init
```

The table below describes the significant fields shown in the display.

Table 24: show memory debug leaks summary Field Descriptions

Field	Description
Alloc_pc	Address of the system call that allocated the block.
Size	Size of the leaked block.
Blocks	Number of blocks leaked.
Bytes	Total amount of memory leaked.
What	Name of the process that owns the block.

Cisco Catalyst 4500e Series Switches running IOS XE software

show memory debug leaks all detailed Command Example

The following example shows output from the **show memory debug leaks all detailed** command:

```
Device# show memory debug leaks all detailed
Process PID : 4644    Process Name : platformmgr
Address   Size    Alloc PC                                          TID         Name
1FEA5E30  20     00000000000000000000000000000000000000000-0+1F5E51BC  4644       XOS_MEM_XDT
```

The table below describes the significant fields shown in the display.

Table 25: show memory debug leaks all detailed Field Descriptions

Field	Description
Address	Hexadecimal address of the leaked block.
Size	Size of the leaked block (in bytes).
Alloc_pc	Address of the system call that allocated the block.
PID	The process identifier of the process that allocated the block
TID	The Task identifier for a particular process identifier.
Name	Name of the process that owns the block.

show memory debug leaks all totals Command Example

The following example shows output from the **show memory debug leaks all totals** command:

```
Device# show memory debug leaks all totals
Process       Num Leak  Total Leak      Num Leak  Total Leak      Total Leak
Name          Mem Blks  Mem Blks(b)     Chunk El  Chunk Elem(b)   (bytes)

slproc        7         232             0         0               232
platformmgr   1         20              0         0               20
ns_oir_prox   8         240             0         0               240
profiled      13        1516            0         0               1516
obfld         5         1464            9         152             1616
consoled      13        1516            0         0               1516
csprovider    13        1552            0         0               1552
system_mgr_   13        1560            0         0               1560
plogd         7         1736            25        968             2704
psdprov       13        1532            0         0               1532
pdsd          16        1564            0         0               1564
gold_slave    9         376             0         0               376
osinfo-prov   13        1576            0         0               1576
oscore_p      13        1520            0         0               1520
netd          10        256             1         28              284
mem_mgmt      82        4620            1         40              4660
mgmte_tap     9         376             0         0               376
licensed      133       16052           95        2660            18712
```

The table below describes the significant fields shown in the display.

Table 26: show memory debug leaks all totals Field Descriptions

Field	Description
Process Name	Name of the leak process.
Num Leak Mem Blks	Number of memory blocks affected by leak for a particular process.
Total Leak Mem Blks(b)	Amount of memory affected by leaks in bytes for a particular process.
Num Leak Chunk El	Number of chunk blocks affected by leak for a particular process.
Total Leak Chunk Elem(b)	Amount of chunk block memory affected by leaks in bytes for a particular process.
Total Leak (bytes)	Total amount of memory leaked for a particular process.

Related Commands

Command	Description
set memory debug incremental starting-time	Sets the current time as the starting time for incremental analysis.
show memory debug incremental allocation	Displays all memory blocks that were allocated after the issue of the **set memory debug incremental starting-time** command.

Command	Description
show memory debug incremental leaks	Displays only memory that was leaked after the issue of the **set memory debug incremental starting-time** command.
show memory debug incremental leaks lowmem	Forces incremental memory leak detection to work in low memory mode. Displays only memory that was leaked after the issue of the **set memory debug incremental starting-time** command.
show memory debug incremental status	Displays if the starting point of incremental analysis has been defined and the time elapsed since then.

show memory debug references

To display debug information on references, use the **show memory debug references**command in user EXEC or privileged EXEC mode.

show memory debug references [**dangling** [*start-address start-address*]]

Syntax Description

dangling	(Optional) Displays the possible references to free memory.
start-address	(Optional) Address numbers <0-4294967295> that determine the address range.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.0	This command was introduced.

Usage Guidelines

All show memory debug commands must be used on customer networks only to diagnose the router for memory leaks when memory depletion is observed. These CLI's will have high CPU utilization and might result in time sensitive protocols to flap. These CLI's are recommended for customer use, only in the maintenance window when the router is not in a scaled condition.

Examples

The following is sample output from the **show memory debug references**command:

```
Router# show memory debug references 2 3
Address  Reference Cont_block Cont_block_name
442850BC        2  44284960   bss
44285110        3  44284960   bss
4429C33C        2  44284960   bss
4429C34C        2  44284960   bss
4429C35C        3  44284960   bss
.
.
.
```

The following is sample output from the **show memory debug references dangling**command:

```
Router# show memory debug references dangling
Address  Reference Free_block Cont_block Cont_block_name
```

```
442D5774 458CE5EC  458CE5BC   44284960   bss
442D578C 46602998  46602958   44284960   bss
442D58A0 465F9BC4  465F9B94   44284960   bss
442D58B8 4656785C  4656781C   44284960   bss
442D5954 45901E7C  45901E4C   44284960   bss
.
.
.
```

The table below describes the significant fields shown in the displays.

Table 27: show memory debug references Field Descriptions

Field	Description
Address	Hexadecimal address of the block having the given or dangling reference.
Reference	Address which is given or dangling.
Free_block	Address of the free block which now contains the memory referenced by the dangling reference.
Cont_block	Address of the control block which contains the block having the reference.
Cont_block_name	Name of the control block.

show memory debug unused

To display debug information on leaks that are accessible, but are no longer needed, use the **show memory debug unused**command in user EXEC or privileged EXEC mode.

show memory debug unused

Syntax Description

This command has no arguments or keywords.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.0	This command was introduced.

Examples

The following is sample output from the **show memory debug unused**command:

```
Router# show memory debug unused
Address  Alloc_pc PID  size     Name
654894B8 62BF31DC -2   44       *Init*
6549A074 601F7A84 -2   4464     XDI data
6549B218 601F7274 -2   4500     XDI data
6549DFB0 6089DDA4 42   84       Init
65509160 6089DDA4 1    84       *Init*
6550A260 6089DDA4 2    84       *Init*
6551FDB4 6089DDA4 4    84       *Init*
6551FF34 627EFA2C -2   24       *Init*
65520B3C 6078B1A4 -2   24       Parser Mode Q1
```

```
65520B88 6078B1C8 -2   24       Parser Mode Q2
65520C40 6078B1A4 -2   24       Parser Mode Q1
65520C8C 6078B1C8 -2   24       Parser Mode Q2
65520D44 6078B1A4 -2   24       Parser Mode Q1
65520D90 6078B1C8 -2   24       Parser Mode Q2
65520E48 6078B1A4 -2   24       Parser Mode Q1
65520E94 6078B1C8 -2   24       Parser Mode Q2
65520F4C 6078B1A4 -2   24       Parser Mode Q1
65520F98 6078B1C8 -2   24       Parser Mode Q2
65521050 6078B1A4 -2   24       Parser Mode Q1
6552109C 6078B1C8 -2   24       Parser Mode Q2
65521154 6078B1A4 -2   24       Parser Mode Q1
655211A0 6078B1C8 -2   24       Parser Mode Q2
.
.
.
```

The table below describes the significant fields shown in the display.

Table 28: show memory debug unused Field Descriptions

Field	Description
Address	Hexadecimal address of the block.
Alloc_pc	Address of the program counter that allocated the block.
PID	Process identifier of the process that allocated the block.
size	Size of the unused block (in bytes).
Name	Name of the process that owns the block.

show memory detailed

To display detailed memory information about POSIX and Cisco IOS processes when Cisco IOS XE or Software Modularity images are running, use the **show memory detailed**command in privileged EXEC mode.

Cisco IOS Software Modularity
show memory detailed [{*process-idprocess-name*}] [{*start-address* [*end-address*] | **bigger** | **free** | **physical** | **shared** | **statistics** | **summary**}]

Cisco Catalyst 4500e Series Switches running IOS XE software
show memory detailed [{**process** {*process-idprocess-name*} | **free** | **io** | **overflow** | **statistics** | **summary**}]

Syntax Description

process-id	(Optional) POSIX process identifier.
process-name	(Optional) POSIX process name.
start-address	(Optional) Starting memory address.
end-address	(Optional) Ending memory address.
bigger	(Optional) Displays information about bigger free blocks in the process.

free	(Optional) Displays free memory information.
io	(Optional) Displays the free I/O memory blocks.
overflow	(Optional) Displays details about memory block header corruption corrections when the **exception memory ignore overflow** global configuration command is configured.
physical	(Optional) Displays physical memory information.
shared	(Optional) Displays shared memory information.
statistics	(Optional) Displays detailed memory usage by address of the system call that allocated the block.
summary	(Optional) Displays summary information about memory usage per system call that allocated the block.

Command Default

No detailed memory information about POSIX and Cisco IOS processes is displayed.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(18)SXF4	This command was introduced to support Software Modularity images.
Cisco IOS XE Release 3.1.0.SG	This command was introduced on the Cisco Catalyst 4500e Serfies Switches.

Usage Guidelines

Detailed output of the process memory on the device is displayed with this command. The process memory summary is displayed first, followed by POSIX and Cisco IOS memory information. The POSIX memory information includes the address, the size in bytes, and the type of memory used by various segments such as program text, data, stack, shared memory, device memory, and heap. Cisco IOS memory information includes the native Cisco IOS display of memory blocks maintained by the Cisco IOS memory management library.

Cisco IOS Software

The following is partial sample output from the **show memory detailed**command for a Cisco IOS process:

```
Router# show memory detailed cdp2.iosproc
System Memory: 131072K total, 115836K used, 15236K free 4000K kernel reserved
Process sbin/cdp2.iosproc, type IOS, PID = 12329
    636K total, 4K text, 4K data, 28K stack, 600K dynamic
    16384 heapsize, 3972 allocated, 10848 free
Address       Bytes What
0x3B42000   4194304 Shared Memory
0x7FBB000      8192 Program Stack
0x8020000     49152 Program Text
0x802C000      4096 Program Data
0x802D000      8192 Allocated memory
0x60000000     4096 Shared Memory "SHM_IDB"
0x60001000    32768 Shared Memory
```

```
             Head       Total(b)  Used(b)    Free(b)   Lowest(b)  Largest(b)
Processor    8034058    508152    480420     27732     17368      18716
          Processor memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
08034058 0000020008 00000000 08038EB8 001  -------- -------- 727FB668 Managed Chunk Queue
Elements
08038EB8 0000002568 08034058 080398F8 001  -------- -------- 72871A44 *Init*
080398F8 0000001512 08038EB8 08039F18 001  -------- -------- 728819D4 List Elements
.
.
.
```

The first section of the display shows system summary information. The table below describes the significant fields shown in the first section of the display.

Table 29: show memory detailed Field Descriptions--First Section

Field	Description
total	Total amount of memory on the device, in kilobytes.
used	Amount of memory in use, in kilobytes.
free	Amount of memory not in use, in kilobytes.
kernel reserved	Amount of memory reserved by the kernel, in kilobytes.

The second section of the display includes process summary statistics about the activities of the system memory allocator. The table below describes the significant fields shown in the second section of the display.

Table 30: show memory detailed Field Descriptions--Second Section

Field	Description
Process	Process name and path.
type	Type of process: POSIX or IOS.
PID	Process ID.
total	Total amount of memory used by the specified process, in kilobytes.
text	Amount of memory, in kilobytes, used by the text segment of the specified process.
data	Amount of memory, in kilobytes, used by the data segment of the specified process.
stack	Amount of memory, in kilobytes, used by the stack segment of the specified process.
dynamic	Amount of memory, in kilobytes, used by the dynamic segment of the specified process.
heapsize	Size of the process heap. Note that the Cisco IOS memory management library allocates heap dynamically. This is shown in the Cisco IOS memory details that follow the POSIX memory display.
allocated	Amount of memory, in kilobytes, allocated from the heap.
free	Amount of free memory, in kilobytes, in the heap for the specified process.

The third section of the display shows POSIX process perspective memory information. The table below describes the significant fields shown in the third section of the display.

Table 31: show memory detailed Field Descriptions--Third Section

Field	Description
Address	Hexadecimal address of block.
Bytes	Size of block (in bytes).
What	Type of memory segment that owns the block, or “(fragment)” if the block is a fragment, or “(coalesced)” if the block was coalesced from adjacent free blocks.

The fourth section of the display shows Cisco IOS memory information as a block-by-block listing of memory use. The table below describes the significant fields shown in the fourth section of the display.

Table 32: show memory detailed Field Descriptions--Fourth Section

Field	Description
Head	Hexadecimal address of the head of the memory allocation chain.
Total(b)	Sum of used bytes plus free bytes.
Used(b)	Amount of memory in use.
Free(b)	Amount of memory not in use.
Lowest(b)	Smallest amount of free memory since last boot.
Largest(b)	Size of largest available free block.
Address	Hexadecimal address of block.
Bytes	Size of block (in bytes).
Prev	Address of previous block (should match address on previous line).
Next	Address of next block (should match address on next line).
PrevF	Address of previous free block (if free).
NextF	Address of next free block (if free).
Alloc PC	Address of the system call that allocated the block.
what	Type of memory segment that owns the block, or “(fragment)” if the block is a fragment, or “(coalesced)” if the block was coalesced from adjacent free blocks.

The following is sample output from the **show memory detailed**command for a POSIX process:

```
Router# show memory detailed 12290
System Memory: 131072K total, 115876K used, 15196K free 4000K kernel reserved
```

```
Process sbin/sysmgr.proc, type POSIX, PID = 12290
    400K total, 100K text, 144K data, 12K stack, 144K dynamic
    81920 heapsize, 68716 allocated, 8824 free
Address       Bytes What
0x7FDF000    126976 Program Stack (pages not allocated)
0x7FFE000      4096 Program Stack
0x8000000    122880 Program Stack (pages not allocated)
0x801E000      8192 Program Stack
0x8020000    102400 Program Text
0x8039000    147456 Program Data
0x805D000      8192 Heap Memory
0x8060000     16384 Heap Memory
0x8064000     16384 Heap Memory
0x8068000      8192 Heap Memory
0x806C000     16384 Heap Memory
0x8070000     16384 Heap Memory
0x8074000     16384 Heap Memory
0x8078000     16384 Heap Memory
0x807C000     16384 Heap Memory
0x8080000     16384 Heap Memory
```

The following partial sample output from the **show memory detailed** command with a process name and the **physical** keyword that displays the summary of physical memory used by the specified process along with the shared memory details:

```
Router# show memory detailed sysmgr.proc physical
Pid       Data   Stack Dynamic    Text  Shared  Maps  Process
20482     304K     16K    256K   3480K    468K   60  sysmgr.proc
Total Physical Memory used or mapped by sysmgr.proc
  Private memory used  (Data/Stack/Dynamic) :      576K
  Shared memory mapped (Text/Shared)        :     3948K
  Number of memory maps                     :       60
Dev   1:Text/Data 2:Mapped 3:Shared 4:DSO
Flags SHD:Shared PRV:Private FXD:Fixed ANN:Anon PHY:Phys
      LZY:Lazy ELF:Elf STK:Stack NOC:Nocache
Phy Addr      Size     Pid   Virt Addr  What    Dev  Prot   MapFlags
0x0          32768K  20482  0x70000000  Text     4  R-X  SHD FXD ELF
0x2000000    32768K  20482  0x72000000  Text     4  R-X  SHD FXD ELF
0x4000000    32768K  20482  0x74000000  Text     4  R-X  SHD FXD ELF
0x522B000        4K  20482  0x1020000   Text     4  R-X  SHD FXD ELF
Phy Addr      Size     Pid   Virt Addr  What    Dev  Prot   MapFlags
0x9EFD4000      32K  20482  0x105C000   Heap     2  RW-  PRV ANN
0x9EFF0000      32K  20482  0x1054000   Heap     2  RW-  PRV ANN
0x9EFF8000      32K  20482  0x1034000   Heap     2  RW-  PRV ANN
0x9F003000       4K  20482  0x7B43C000  Data     4  RW-  PRV FXD ANN ELF
.
.
.
```

The table below describes the significant fields shown in the display.

Table 33: show memory detailed Field Descriptions

Field	Description
Shared	Amount of memory shared by the specified process, in kilobytes.
Maps	Number of memory maps for the specified process.
Process	Name of the process.
Private memory used	Total amount of private memory used by the process.

Field	Description
Shared memory mapped	Total amount of shared memory used by the process.
Number of memory maps	Total number of maps for the process.
Flags	Flags that specify information about handling of the mapped region. The available flags are as follows: • SHD:Shared--Specifies that memory is shared between different process. • PRV:Private--Specifies that memory is private to this process. • FXD:Fixed--Specifies that memory is mapped to a fixed virtual address in the process. • ANN:Anon--Specifies that physical memory was allocated by the kernel. • PHY:Phys--Specifies that the user specified the physical memory. • LZY:Lazy--Specifies that memory is lazy mapped; that is, physical memory is not allocated until the memory is either read or written to other memory. • ELF:Elf--Specifies that memory is an Executable and Linkable Format (ELF) object. • STK:Stack--Specifies that memory is used for stack. • NOC:Nocache--Specifies that memory is set up without any cache.
Phy Addr	Hexadecimal address of the physical memory block.
Size	Amount of physical memory mapped in the process of development.
Virt Addr	Virtual memory to which this memory is mapped.
Prot	Memory protection settings for the memory--read, write, and execute.
MapFlags	Represents special mapping properties used for the memory.

Cisco Catalyst 4500e Series Switches running IOS XE software

The following is sample output from the **show memory detailed** command for the *iosd* process:

```
Switch#show memory detailed proc iosd
System memory  : 883144K total, 591378K used, 291766K free, 165432K kernel reserved
Lowest(b)      : 5128192
Process iosd, type L, PID = 11007
    777572K total, 82212K text, 537120K data, 84K stack, 240K dynamic
    240 heapsize, 240 allocated, 0 free
               Head    Total(b)     Used(b)     Free(b)   Lowest(b)  Largest(b)
Processor   90150008   536870912   261852128   275018784   273655520   272592492
      I/O   B0151000    16777216      169288    16607928    16598952    16598948
          Processor memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
90150008 0000000436 00000000 901501E8 001  -------- -------- 1028C010  *Init*
901501E8 0000020004 90150008 90155038 001  -------- -------- 11D5E9D4  Managed Chunk Queue
```

```
 Elements
90155038 0000065540 901501E8 90165068 001  -------- -------- 11D5F518  MallocLite
90165068 0000065540 90155038 90175098 001  -------- -------- 11D5F518  MallocLite
90175098 0000065540 90165068 901850C8 001  -------- -------- 11D5F518  MallocLite
901850C8 0000065540 90175098 901950F8 001  -------- -------- 11D5F518  MallocLite
901950F8 0000000524 901850C8 90195330 001  -------- -------- 1028C5C4  *Init*
90195330 0000065540 901950F8 901A5360 001  -------- -------- 11D5F518  MallocLite
901A5360 0000002620 90195330 901A5DC8 001  -------- -------- 1028C770  *Init*
901A5DC8 0000000892 901A5360 901A6170 001  -------- -------- 12A39D50  *Init*
901A6170 0000000892 901A5DC8 901A6518 001  -------- -------- 12A39D50  *Init*
901A6518 0000131076 901A6170 901C6548 001  -------- -------- 12A3A154  *Init*
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
901C6548 0000065540 901A6518 901D6578 001  -------- -------- 11D5F518  MallocLite
901D6578 0000000956 901C6548 901D6960 001  -------- -------- 11445508  IPC Seat
901D6960 0000000404 901D6578 901D6B20 001  -------- -------- 1107D218  Exec
901D6B20 0000000092 901D6960 901D6BA8 001  -------- -------- 110533B0  TTYBKG Timer
901D6BA8 0000000684 901D6B20 901D6E80 001  -------- -------- 0CCA9660  SPI PL client app
handler
901D6E80 0000000148 901D6BA8 901D6F40 001  -------- -------- 0CCA9660  SPI PL client app
handler
901D6F40 0000064252 901D6E80 901E6A68 000  9ED89128 0        13A89380  (coalesced)
901E6A68 0000080004 901D6F40 901FA318 001  -------- -------- 0CCA9660  SL async process
901FA318 0000002068 901E6A68 901FAB58 001  -------- -------- 110796B0  Exec
901FAB58 0000001108 901FA318 901FAFD8 000  9FB2D988 0        110796B0  (fragment)
901FAFD8 0000064100 901FAB58 9020AA68 001  -------- -------- 10B6D078  Process Stack
9020AA68 0001286420 901FAFD8 90344BA8 000  9FD59170 0        10B6D078  (fragment)
90344BA8 0000012804 9020AA68 90347DD8 001  -------- -------- 13A96844  *Init*
 --More-- [nova-k5-14:~]$ ioucon 100
          I/O memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
B0151000 0000000260 00000000 B0151130 001  -------- -------- 10519010  *Packet Data*
B0151130 0000000260 B0151000 B0151260 001  -------- -------- 10519010  *Packet Data*
B0151260 0000000260 B0151130 B0151390 001  -------- -------- 10519010  *Packet Data*
B0151390 0000000260 B0151260 B01514C0 001  -------- -------- 10519010  *Packet Data*
B01514C0 0000000260 B0151390 B01515F0 001  -------- -------- 10519010  *Packet Data*
B01515F0 0000000260 B01514C0 B0151720 001  -------- -------- 10519010  *Packet Data*
B0151720 0000000260 B01515F0 B0151850 001  -------- -------- 10519010  *Packet Data*
B0151850 0000000260 B0151720 B0151980 001  -------- -------- 10519010  *Packet Data*
B0151980 0000000260 B0151850 B0151AB0 001  -------- -------- 10519010  *Packet Data*
Switch#
```

Related Commands

Command	Description
show memory	Displays system memory information.
show memory detailed all	Displays detailed memory information of all applicable processes.

show memory ecc

To display single-bit Error Code Correction (ECC) error logset data, use the show memory ecc command in privileged EXEC mode.

show memory ecc

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.1(30)CC	This command was introduced in Cisco IOS Release 11.1(30)CC.
12.0(4)XE	This command was integrated into Cisco IOS Release 12.0(4)XE.
12.0(6)S	This command was integrated into Cisco IOS Release 12.0(6)S.
12.1(13)	This command was integrated into Cisco IOS Release 12.1(13).

Usage Guidelines

Use this command to determine if the router has experienced single-bit parity errors.

Examples

The following is sample output from the show memory ecc command from a 12000-series router running Cisco IOS Release 12.0(23)S:

```
Router# show memory ecc
ECC Single Bit error log
------------------------
Single Bit error detected and corrected at 0x574F3640
- Occured 1 time(s)
- Whether a scrub was attempted at this address: Yes
- Syndrome of the last error at this address: 0xE9
- Error detected on a read-modify-write cycle ? No
- Address region classification: Unknown
- Address media classification : Read/Write Single Bit error detected and corrected at
0x56AB3760
- Occured 1 time(s)
- Whether a scrub was attempted at this address: Yes
- Syndrome of the last error at this address: 0x68
- Error detected on a read-modify-write cycle ? No
- Address region classification: Unknown
- Address media classification : Read/Write
Total Single Bit error(s) thus far: 2
```

The table below describes the significant fields shown in the first section of the display.

Table 34: show memory ecc Field Descriptions

Field	Description
Occured n time(s)	Number of single-bit errors that has occurred.
Whether a scrub was attempted at this address:	Indicates whether a scrub has been performed.
Syndrome of the last error at this address:	Describes the syndrome of last error.
Error detected on a read-modify-write cycle ?	Indicates whether an error has occurred.
Address region classification: Unknown	Describes the region of the error.
Address media classification :	Describes the media of the error and correction.

Related Commands

Command	Description
show memory	Displays statistics about memory, including memory-free pool statistics.

show memory events

To display recorded memory events, use the **show memory events**command in privileged EXEC mode.

show memory events [**outstanding** [**summary**]]

Syntax Description

outstanding	(Optional) Displays the outstanding allocation events in the event buffer.
summary	(Optional) Displays a summary of outstanding allocation events in the event buffer.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.

Usage Guidelines

Before you can enable the **show memory events** command, you must configure the **memory record events** command in global configuration mode.

Examples

The following is sample output from the **show memory events**command:

```
Router# configure terminal

Router(config)# memory record events
Memory event recording already enabled!
Router(config)# exit
Router# show memory events
Last recorded memory events:
When         Type Block/Chunk DataPtr Size PID What Traceback/PC
4d19h        FREE 695B3200 695B3230 3000   82 Iterator Hash Entry 615B75C4
```

The table below describes the significant fields shown in the display.

Table 35: show memory events Field Descriptions

Field	Description
When	Time when the memory event was last seen by the system (in hours and days).
Type	Allocation type.
Block/Chunk/DataPtr	Number of memory events allocated.
Size	Amount of memory, in bytes, used by the task.
PID	Packet identification number.
What	Name of the process that owns a block or fragment.
Traceback/PC	Traceback error.

The following is sample output from the **show memory events**command using the **outstanding and summary**keywords:

```
Router# configure terminal
Router(config)# memory record events
Memory event recording already enabled!
Router(config)# exit
Router# show memory events outstanding summary

Last-Seen    Type   How-Many Size     PID What         Traceback/PC
5d16h        ALLOC    1        320    135 Exec               61B399F4
```

The table below describes the significant fields shown in the display.

Table 36: show memory events Field Descriptions

Field	Description
Last-Seen	Time when the memory event was last seen by the system (in hours and days).
Type	Allocation type.
How-Many	Number of memory events allocated.
Size	Amount of memory, in bytes, used by the task.
PID	Packet identification number.
What	Name of the process that owns a block or fragment.
Traceback/PC	Traceback error.

Related Commands

Command	Description
show memory traceback	Displays memory traceback information.

show memory failures alloc

To display statistics about failed memory allocation requests, use the **show memory failures alloc**command in the privileged EXEC mode.

show memory failures alloc

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.0	This command was introduced.

Examples

The following is sample output from the **show memory failures alloc** command:

```
Router# show memory failures alloc

Caller       Pool          Size   Alignment   When
0x60394744  I/O            1684     32      00:10:03
0x60394744  I/O            1684     32      00:10:03
0x60394744  I/O            1684     32      00:10:03
0x60394744  I/O            1684     32      00:10:03
0x60394744  I/O            1684     32      00:10:03
0x60394744  I/O            1684     32      00:10:03
0x60394744  I/O            1684     32      00:10:03
0x60394744  I/O            1684     32      00:10:03
0x60394744  I/O            1684     32      00:10:04
0x60394744  I/O            1684     32      00:10:04
```

The table below describes the significant fields shown in the display.

Table 37: show memory failures alloc Field Descriptions

Field	Description
Caller	Address of the allocator function that issued memory allocation request that failed.
Pool	Pool from which the memory was requested.
Size	Size of the memory requested in bits.
Alignment	Memory alignment in bits.
When	Time of day at which the memory allocation request was issued.

show memory fast

To display fast memory details for the router, use the **show memory fast** command.

show memory fast [{**allocating-process** [**totals**] | **dead** [**totals**] | **free** [**totals**]}]

Syntax Description

allocating-process	(Optional) Include allocating process names with the standard output.
dead	(Optional) Display only memory owned by dead processes.
free	(Optional) Display only memory not allocated to a process.
totals	(Optional) Summarizes the statistics for allocating processes, dead memory, or free memory.

Command Modes

Exec

Command History

Release	Modification
12.1	This command was introduced in a release prior to 12.1. This command replaced the **show memory sram** command.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The show memory fast command displays the statistics for the fast memory. “Fast memory” is another name for “processor memory,” and is also known as “cache memory.” Cache memory is called fast memory because the processor can generally access the local cache (traditionally stored on SRAM positioned close to the processor) much more quickly than main memory or RAM.

Note The **show memory fast** command is a command alias for the **show memory processor** command. These commands will issue the same output.

Examples

The following example shows sample output from the **show memory fast** and the **show memory processor** commands:

```
Router>show memory fast

          Processor memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
8404A580 0001493284 00000000 841B6ECC 000  0        84BADF88 815219D8  (coalesced)
841B6ECC 0000020004 8404A580 841BBD18 001  -------- -------- 815DB094  Managed Chunk Queue
 Elements
841BBD18 0000001504 841B6ECC 841BC320 001  -------- -------- 8159EAC4  List Elements
841BC320 0000005004 841BBD18 841BD6D4 001  -------- -------- 8159EB04  List Headers
841BD6D4 0000000048 841BC320 841BD72C 001  -------- -------- 81F2A614  *Init*
841BD72C 0000001504 841BD6D4 841BDD34 001  -------- -------- 815A9514  messages
841BDD34 0000001504 841BD72C 841BE33C 001  -------- -------- 815A9540  Watched messages
841BE33C 0000001504 841BDD34 841BE944 001  -------- -------- 815A95E4  Watched Semaphore
841BE944 0000000504 841BE33C 841BEB64 001  -------- -------- 815A9630  Watched Message Queue
841BEB64 0000001504 841BE944 841BF16C 001  -------- -------- 815A9658  Watcher Message Queue
841BF16C 0000001036 841BEB64 841BF5A0 001  -------- -------- 815A2B24  Process Array
-- More --
<Ctrl+z>
Router>show memory processor

          Processor memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
8404A580 0001493284 00000000 841B6ECC 000  0        84BADF88 815219D8  (coalesced)
841B6ECC 0000020004 8404A580 841BBD18 001  -------- -------- 815DB094  Managed Chunk Queue
 Elements
841BBD18 0000001504 841B6ECC 841BC320 001  -------- -------- 8159EAC4  List Elements
841BC320 0000005004 841BBD18 841BD6D4 001  -------- -------- 8159EB04  List Headers
841BD6D4 0000000048 841BC320 841BD72C 001  -------- -------- 81F2A614  *Init*
841BD72C 0000001504 841BD6D4 841BDD34 001  -------- -------- 815A9514  messages
841BDD34 0000001504 841BD72C 841BE33C 001  -------- -------- 815A9540  Watched messages
841BE33C 0000001504 841BDD34 841BE944 001  -------- -------- 815A95E4  Watched Semaphore
841BE944 0000000504 841BE33C 841BEB64 001  -------- -------- 815A9630  Watched Message Queue
841BEB64 0000001504 841BE944 841BF16C 001  -------- -------- 815A9658  Watcher Message Queue
841BF16C 0000001036 841BEB64 841BF5A0 001  -------- -------- 815A2B24  Process Array
-- More --
<Ctrl+z>
Router>
```

The following example shows sample output from the **show memory fast allocating-process** command, followed by sample output from the **show memory fast allocating-process totals** command:

```
Router#show memory fast allocating-process

          Processor memory
```

```
 Address      Bytes     Prev     Next Ref   Alloc Proc     Alloc PC  What
8404A580 0001493284 00000000 841B6ECC 000                  815219D8  (coalesced)
841B6ECC 0000020004 8404A580 841BBD18 001  *Init*         815DB094  Managed Chunk Queue
Elements
841BBD18 0000001504 841B6ECC 841BC320 001  *Init*         8159EAC4  List Elements
841BC320 0000005004 841BBD18 841BD6D4 001  *Init*         8159EB04  List Headers
841BD6D4 0000000048 841BC320 841BD72C 001  *Init*         81F2A614  *Init*
841BD72C 0000001504 841BD6D4 841BDD34 001  *Init*         815A9514  messages
841BDD34 0000001504 841BD72C 841BE33C 001  *Init*         815A9540  Watched messages
841BE33C 0000001504 841BDD34 841BE944 001  *Init*         815A95E4  Watched Semaphore
841BE944 0000000504 841BE33C 841BEB64 001  *Init*         815A9630  Watched Message Queue
841BEB64 0000001504 841BE944 841BF16C 001  *Init*         815A9658  Watcher Message Queue
841BF16C 0000001036 841BEB64 841BF5A0 001  *Init*         815A2B24  Process Array
 --More--
<Ctrl+z>
c2600-1#show memory fast allocating-process totals

Allocator PC Summary for: Processor
    PC          Total   Count  Name
0x815C085C    1194600     150  Process Stack
0x815B6C28     948680       5  pak subblock chunk
0x819F1DE4     524640       8  BGP (0) update
0x815C4FD4     393480       6  MallocLite
0x815B5FDC     351528      30  TW Buckets
0x819F14DC     327900       5  connected
0x81A1E838     327900       5  IPv4 Unicast net-chunk(8)
0x8153DFB8     248136     294  *Packet Header*
0x82142438     133192       4  CEF: 16 path chunk pool
0x82151E0C     131116       1  Init
0x819F1C8C     118480       4  BGP (0) attr
0x815A4858     100048     148  Process
0x8083DA44      97248      17
 --More--
<Ctrl+z>
```

The following example shows sample output from the **show memory fast dead** command:

```
Router#show memory fast dead

          Processor memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
8498FC20 0000000028 8498FB90 8498FC64 001  -------- -------- 81472B24  AAA MI SG NAME
        -------
             68
Router#show memory fast dead totals

Dead Proc Summary for: Processor
    PC          Total   Count  Name
0x81472B24         68       1  AAA MI SG NAME
Router#
```

show memory fragment

To display the block details of fragmented free blocks and allocated blocks, which is physically just before or after the blocks on the free list, use the **show memory fragment**command in user EXEC or privileged EXEC mode.

show memory [{**processor** | **io**}] **fragment** [**detail**]

Syntax Description

processor	(Optional) Displays the processor memory information.
io	(Optional) Displays the I/O memory information.
fragment	Displays the information of the free blocks and the blocks surrounding the free blocks.
detail	(Optional) Displays the detailed information of all the free blocks and the blocks surrounding the free blocks that are located between the allocated blocks.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

Examples

The following is sample output from the **show memory processor fragment** command:

```
Router# show memory processor fragment

Processor memory
Free memory size : 65516944 Number of free blocks:      230
Allocator PC Summary for allocated blocks in pool: Processor
    PC          Total   Count  Name
0x6047DDCC     852020       1  atmdx_vc_table
0x6075DC30     544392       4  ATM1/0
0x61BDBA14     131176       2  eddri_self_event
0x61913BEC     131124       1  l2tp tnl table
0x602E9820     114832       1  AutoVC Msg Chunk
0x6071253C      98408       2  Exec
0x607DF5BC      96624      12  Process Stack
0x6118DDA0      77252       1  Spanning Tree Opt Port Block
0x61F13C30      67636       1  QOS_MODULE_MAIN
0x6047DD3C      65640       2  atmdx_tx_shadow
0x614B6624      65588       1  CEF: loadinfo chunk
0x614D1924      65588       1  IP mtrie node
0x614A58A0      65588       1  CEF: 16 path chunk pool
0x619241D4      65588       1  PPTP mgd timer chunk
0x606581CC      65588       1  AAA DB Chunk
0x607E5EAC      65588       1  MallocLite
0x6192420C      65588       1  PPTP: pptp_tunneltype chunk
0x6075DCB8      45924      10  FastEthernet2/
0x607CA400      36288       2  pak subblock chunk
0x6255648C      28948       1  CCPROXY_CT
0x6047DD7C      24628       1  atmdx_bfd_cache
0x6047DAA4      23500       1  atmdx_instance
0x6047DAE8      23500       1  atmdx_instance snap
0x60962DFC      21420      17  TCP CB
0x616F729C      20052       1  AC context chunks
0x616F72C8      20052       1  AC Mgr mgd timer chunk
0x60734010      16644      19  *Packet Header*
0x6047DE0C      16436       1  atmdx_abr_stats
0x6047DCFC      16112       2  atmdx_rx_pool_info
0x60A77E98      13060       1  DHCPD Message Workspace
0x61F50008      12852       1  CCVPM_HTSP
0x60D509BC      12580      17  Virtual Exec
0x60EFA1EC      12344       1  RSVP DB Handle Bin
```

```
.
.
.
0x6067AE44         76       1  AAA Secrettype encrypt
0x61C0EEC0         76       1  Init
0x60F76B1C         76       1  SNMP Trap
0x60BE2444         76       1  Init
0x62638F78         76       1  EEM ED Syslog
0x6077C574         76       1  Init
0x608F7030         76       1  IPC Name String
0x608EEAB8         76       1  IPC Name
0x620468A8         76       1  ivr: ccapAppEntry_t name
0x6066D084         76       1  gk process
0x6064824C         76       1  AAA MI SG NAME
Allocator PC Summary for free blocks in pool: Processor
    PC          Total   Count  Name
0x6071253C   67387912       2  (fragment)
0x60734010   63292440      11  *Packet Header*
0x60962DFC     105552      10  (coalesced)
0x60D509BC      98384      10  (coalesced)
0x60D4A0B4      70776       9  (coalesced)
0x60803260      21488       4  (fragment)
0x60B2E488      19704       2  (fragment)
0x606E0278      19272       1  (coalesced)
0x606DD8D8       9024     113  Init
0x60B27FE8       5740       3  (fragment)
0x60778AAC       3504       1  (coalesced)
0x607AC764       2212      11  Process Events
0x60F7FCD4       1556       9  (fragment)
0x6071F3FC       1316      12  (fragment)
0x606C5324       1176       6  (coalesced)
0x60D7C518       1148       1  (coalesced)
0x624E170C        876       1  (coalesced)
0x60A68164        588       3  (fragment)
0x60B302C0        408       5  (fragment)
0x60976574        272       2  AAA Event Data
0x60801E38        216       2  (fragment)
0x611DA23C        164       1  shelf_info
0x60A6A638        148       1  (fragment)
0x60801D2C        148       1  (fragment)
0x60D29DCC        148       1  (fragment)
0x62628CA0        144       1  (fragment)
0x60A68218        104       1  (fragment)
0x606B9614         88       1  NameDB String
0x6090A978         84       1  (fragment)
0x606C51D0         84       1  (fragment)
0x62647558         76       1  (fragment)
```

The following is sample output from the **show memory processor fragment detail**command:

```
Router# show memory processor fragment detail

Processor memory
Free memory size : 65566148 Number of free blocks:      230
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
645A8148 0000000028 645A80F0 645A8194 001  -------- -------- 60695B20  Init
645A8194 0000000040 645A8148 645A81EC 000  0        200B4300 606B9614  NameDB String
645A81EC 0000000260 645A8194 645A8320 001  -------- -------- 607C2D20  Init
200B42B4 0000000028 200B4268 200B4300 001  -------- -------- 62366C80  Init
200B4300 0000000028 200B42B4 200B434C 000  645A8194 6490F7E8 60976574  AAA Event Data
200B434C 0000002004 200B4300 200B4B50 001  -------- -------- 6267D294  Coproc Request
Structures
6490F79C 0000000028 6490F748 6490F7E8 001  -------- -------- 606DDA04  Parser Linkage
6490F7E8 0000000028 6490F79C 6490F834 000  200B4300 6491120C 606DD8D8  Init
```

```
6490F834 0000006004 6490F7E8 64910FD8 001  -------- -------- 607DF5BC  Process Stack
649111A0 0000000060 64911154 6491120C 001  -------- -------- 606DE82C  Parser Mode
6491120C 0000000028 649111A0 64911258 000  6490F7E8 500770F0 606DD8D8  Init
64911258 0000000200 6491120C 64911350 001  -------- -------- 603F0E38  Init
.
.
.
504DCF54 0000001212 504DB2E4 504DD440 001  -------- -------- 60962DFC  TCP CB
2C41DCA4 0000000692 2C41BCC8 2C41DF88 001  -------- -------- 60D509BC  Virtual Exec
2C41DF88 0000005344 2C41DCA4 2C41F498 000  504DB2E4 6449A828 60D509BC  (coalesced)
2C41F498 0000000692 2C41DF88 2C41F77C 001  -------- -------- 60D509BC  Virtual Exec
6449A544 0000000692 64499794 6449A828 001  -------- -------- 60D509BC  Virtual Exec
6449A828 0000007760 6449A544 6449C6A8 000  2C41DF88 504D89D4 60D509BC  (coalesced)
6449C6A8 0000008044 6449A828 6449E644 001  -------- -------- 60D2AACC  Virtual Exec
504D8778 0000000556 504D754C 504D89D4 001  -------- -------- 60D4A0B4  Virtual Exec
504D89D4 0000009860 504D8778 504DB088 000  6449A828 504D1B78 60D4A0B4  (coalesced)
504DB088 0000000556 504D89D4 504DB2E4 001  -------- -------- 60D4A0B4  Virtual Exec
504D168C 0000001212 504C9658 504D1B78 001  -------- -------- 60962DFC  TCP CB
504D1B78 0000008328 504D168C 504D3C30 000  504D89D4 504C5B54 60962DFC  (coalesced)
504D3C30 0000001212 504D1B78 504D411C 001  -------- -------- 60962DFC  TCP CB
504C5870 0000000692 504C5504 504C5B54 001  -------- -------- 60D509BC  Virtual Exec
504C5B54 0000005344 504C5870 504C7064 000  504D1B78 2C423A88 60D509BC  (coalesced)
504C7064 0000000408 504C5B54 504C722C 001  -------- -------- 606E0E44  Chain Cache No
2C42359C 0000001212 2C41F77C 2C423A88 001  -------- -------- 60962DFC  TCP CB
2C423A88 0000008328 2C42359C 2C425B40 000  504C5B54 504D411C 60962DFC  (coalesced)
504E7DD8 0000000828 504E2660 504E8144 001  -------- -------- 60734010  *Packet Header*
65006A08 0000000408 65003834 65006BD0 001  -------- -------- 606E0E44  Chain Cache No
65006BD0 0000020520 65006A08 6500BC28 000  504E2660 0        60803260  (coalesced)
6500BC28 0000000828 65006BD0 6500BF94 001  -------- -------- 60734010  *Packet Header*
5C3AE7B8 0000000828 5C3AE614 5C3AEB24 001  -------- -------- 60734010  *Packet Header*
5C3AEB24 0063247532 5C3AE7B8 20000000 000  0        6500C300 60734010  (coalesced)
20000000 0000000828 5C3AEB24 2000036C 001  -------- -------- 60734010  *Packet Header*
6500BF94 0000000828 6500BC28 6500C300 001  -------- -------- 60734010  *Packet Header*
6500C300 0004760912 6500BF94 50000000 000  5C3AEB24 2C42E310 6071253C  (coalesced)
50000000 0000000828 6500C300 5000036C 001  -------- -------- 60734010  *Packet Header*
2C42E0B4 0000000556 2C429430 2C42E310 001  -------- -------- 60D4A0B4  Virtual Exec
2C42E310 0062725312 2C42E0B4 00000000 000  6500C300 0        6071253C  (coalesced
```

Related Commands

Command	Description
memory io	Configures thresholds for I/O memory.
memory processor	Configures thresholds for processor memory.

show memory lite-chunks

To display statistics about malloc-lite memory, use the **show memory lite-chunks** command in user EXEC or privileged EXEC mode

show memory lite-chunks {**statistics** | **totals**} {**summary**{**pool** | {**all***pool*}}}

Syntax Description

statistics	Displays malloc lite utilization statistics sorted by pool.
totals	Displays malloc lite allocating totals.
summary	Displays a summary of malloc lite usage for all or a specific pool.

pool	Displays malloc lite allocation information for all pools or for a specific pool.
all	Displays malloc lite information for all pools.
pool	Name of a specific pool.

Command Modes

User EXEC
Privileged EXEC

Command History

Release	Modification
15.0T	This command was introduced.

Usage Guidelines

Example

The following is sample output from the **show memory lite-chunks** command.

```
Device# show memory lite-chunks pool 8

       8 bytes pool

  Address   Ref   Alloc PC
 69D0CCBC   000   64286AAC
 69D0CCD8   000   64286AAC
 69D0CCF4   000   64286AAC
 69D0CD10   000   64286AAC
 69D0CD2C   000   64286AAC
 69D0CD48   000   64286AAC
 69D0CD64   000   64286AAC
 69D0CD80   000   64286AAC
 69D0CD9C   000   64286AAC
 69D0CDB8   000   64286AAC
 69D0CDD4   000   64286AAC
 69D0CDF0   000   64286AAC
 69D0CE0C   000   64286AAC
 69D0CE28   000   64286AAC
 69D0CE44   000   64286AAC
 69D0CE60   000   64286AAC
 69D0CE7C   000   64286AAC
 69D0CE98   000   64286AAC
 69D0CEB4   000   64286AAC
```

The table below describes the significant fields shown in the display.

Table 38: show memory lite-chunks Field Descriptions

Field	Description
Address	Hexadecimal address of the block.
Ref	Reference count for that memory block, indicating how many different processes are using that block of memory.
Alloc PC	Address of the program counter that allocated the block.

show memory multibus

To display statistics about multibus memory, including memory-free pool statistics, use the **show memory multibus**command in user EXEC or privileged EXEC mode.

show memory multibus [{**allocating-process** [**totals**] | **dead** [**totals**] | **free** [**totals**]}]

Syntax Description

allocating-process [totals]	(Optional) Displays allocating memory totals by name.
dead [totals]	(Optional) Displays memory totals on dead processes.
fragment **[detail]**	(Optional) Displays memory statistics for fragmented processes.
free [totals]	(Optional) Displays statistics on free memory.
statistics **[history]**	(Optional) Displays memory pool history statistics on all processes.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.0	This command was introduced.

Examples

The following is sample output from the **show memory multibus** command:

```
Router# show memory multibus
          Processor memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
6540BBA0 0000016388 00000000 6540FBD4 001  -------- -------- 60883984  TW Buckes
6540FBD4 0000016388 6540BBA0 65413C08 001  -------- -------- 60883984  TW Buckes
65413C08 0000016388 6540FBD4 65417C3C 001  -------- -------- 60883984  TW Buckes
65417C3C 0000006004 65413C08 654193E0 001  -------- -------- 608A0D4C  Process k
654193E0 0000012004 65417C3C 6541C2F4 001  -------- -------- 608A0D4C  Process k
6541C2F4 0000411712 654193E0 65480B64 000  0        0        608A0D4C  (fragmen)
65480B64 0000020004 6541C2F4 654859B8 001  -------- -------- 608CF99C  Managed s
654859B8 0000010004 65480B64 654880FC 001  -------- -------- 6085C7F8  List Eles
654880FC 0000005004 654859B8 654894B8 001  -------- -------- 6085C83C  List Heas
654894B8 0000000048 654880FC 65489518 001  -------- -------- 62BF31DC  *Init*
.
.
.
```

The table below describes the significant fields shown in the display.

Table 39: show memory multibus Field Descriptions

Field	Description
Address	Hexadecimal address of the block.
Bytes	Size of the block (in bytes).
Prev	Address of the preceding block (should match the address on the preceding line).

Field	Description
Next	Address of the following block (should match the address on the following line).
Ref	Reference count for that memory block, indicating how many different processes are using that block of memory.
PrevF	Address of the preceding free block (if free).
NextF	Address of the following free block (if free).
Alloc PC	Address of the program counter that allocated the block.
What	Name of the process that owns the block, or "(fragmen)" if the block is a fragment, or "(coalesced)" if the block was coalesced from adjacent free blocks.

show memory pci

To display statistics about Peripheral Component Interconnect (PCI) memory, use the **show memory pci** command in user EXEC or privileged EXEC mode.

show memory pci

Syntax Description

This command has no arguments or keywords.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.0	This command was introduced.

Examples

The following is sample output from the **show memory pci** command:

```
Router# show memory pci
          I/O memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
0E000000 0000000032 00000000 0E000050 000  64F5EBF4 0        00000000  (fragmen)
0E000050 0000000272 0E000000 0E000190 001  -------- -------- 607E2EC0  *Packet *
0E000190 0000000272 0E000050 0E0002D0 001  -------- -------- 607E2EC0  *Packet *
0E0002D0 0000000272 0E000190 0E000410 001  -------- -------- 607E2EC0  *Packet *
0E000410 0000000272 0E0002D0 0E000550 001  -------- -------- 607E2EC0  *Packet *
0E000550 0000000272 0E000410 0E000690 001  -------- -------- 607E2EC0  *Packet *
0E000690 0000000272 0E000550 0E0007D0 001  -------- -------- 607E2EC0  *Packet *
0E0007D0 0000000272 0E000690 0E000910 001  -------- -------- 607E2EC0  *Packet *
0E000910 0000000272 0E0007D0 0E000A50 001  -------- -------- 607E2EC0  *Packet *
0E000A50 0000000272 0E000910 0E000B90 001  -------- -------- 607E2EC0  *Packet *
0E000B90 0000000272 0E000A50 0E000CD0 001  -------- -------- 607E2EC0  *Packet *
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
0E000CD0 0000000272 0E000B90 0E000E10 001  -------- -------- 607E2EC0  *Packet *
0E000E10 0000000272 0E000CD0 0E000F50 001  -------- -------- 607E2EC0  *Packet *
```

The table below describes the significant fields shown in the display.

Table 40: show memory pci Field Descriptions

Field	Description
Address	Hexadecimal address of the block.
Bytes	Size of the block (in bytes).
Prev	Address of the preceding block (should match the address on the preceding line).
Next	Address of the following block (should match the address on the following line).
Ref	Reference count for that memory block, indicating how many different processes are using that block of memory.
PrevF	Address of the preceding free block (if free).
NextF	Address of the following free block (if free).
Alloc PC	Address of the program counter that allocated the block.
what	Name of process that owns the block, or “(fragmen)” if the block is a fragment, or "(coalesced)" if the block was coalesced from adjacent free blocks.

show memory processor

To display statistics on the Router Processor memory, use the **show memory processor** command in user EXEC or privileged EXEC mode.

show memory processor [{**allocating-process** [**totals**] | **dead** [**totals**] | **fragment** [**detail**] | **free** [**totals**] | **statistics**}]

Syntax Description

allocating-process	(Optional) Displays the allocating process name.
totals	(Optional) Displays the total allocated memory.
dead	(Optional) Displays information about memory owned by dead processes.
totals	(Optional) Displays the total dead process memory.
fragment	(Optional) Displays the block details of fragmented free blocks and allocated blocks, which are shown either preceding or following the blocks on the free list.
detail	(Optional) Displays memory fragment information in detail.
free	(Optional) Displays the statistics of the available processor memory.
totals	(Optional) Displays the total free memory.
statistics	(Optional) Displays memory pool statistics.

Command Modes

User EXEC (>) Privileged EXEC (#)

Command History

Release	Modification
12.0	This command was introduced.
12.4(24)T	This command was modified in a release earlier than Cisco IOS Release 12.4(24)T. The **allocating-process** and **dead** keywords were added.

Examples

The following is sample output from the **show memory processor**command:

```
Router# show memory processor
          Processor memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
6540BBA0 0000016388 00000000 6540FBD4 001  -------- -------- 60883984  TW Buckes
6540FBD4 0000016388 6540BBA0 65413C08 001  -------- -------- 60883984  TW Buckes
65413C08 0000016388 6540FBD4 65417C3C 001  -------- -------- 60883984  TW Buckes
65417C3C 0000006004 65413C08 654193E0 001  -------- -------- 608A0D4C  Process k
654193E0 0000012004 65417C3C 6541C2F4 001  -------- -------- 608A0D4C  Process k
6541C2F4 0000411712 654193E0 65480B64 000  0        0        608A0D4C  (fragmen)
65480B64 0000020004 6541C2F4 654859B8 001  -------- -------- 608CF99C  Managed s
654859B8 0000010004 65480B64 654880FC 001  -------- -------- 6085C7F8  List Eles
654880FC 0000005004 654859B8 654894B8 001  -------- -------- 6085C83C  List Heas
654894B8 0000000048 654880FC 65489518 001  -------- -------- 62BF31DC  *Init*
```

The table below describes the significant fields shown in the display.

Table 41: show memory processor Field Descriptions

Field	Description
Address	Hexadecimal address of the block.
Bytes	Size of the block (in bytes).
Prev	Address of the preceding block (should match the address on the preceding line).
Next	Address of the following block (should match the address on the following line).
Ref	Reference count for that memory block, indicating how many different processes are using that block of memory.
PrevF	Address of the preceding free block (if free).
NextF	Address of the following free block (if free).
Alloc PC	Address of the program counter that allocated the block.
What	Name of the process that owns the block or fragment.

The following is sample output from the **show memory processor allocating-process**command:

```
Router# show memory processor allocating-process
PC               Total   Count      Name
0x6013A948    3719220       1  atmdx_setup_vc_table
0x6064EB28    2581132     291  Process Stack
0x627E2420    2569476      78  CCE dp subbloc
0x62A098C8    1637116      24  regex
```

```
0x62EAF010     979876      77  TW Buckets
0x602439EC     935064     962  *Packet Header*
0x614B3A4C     916724      13  Init
0x6013A89C     852020       1  atmdx_vc_table
0x61A54AEC     786292       1  Init
0x62D7BDD0     702336     160  TCL Chunks
0x62EB0458     666988      14  pak subblock chunk
0x60767C38     641076       1  CCPROXY_CT
0x607439C4     524340       1  L2X Hash Table
0x60271864     434328      28  Normal
0x602718F8     407592     148  Normal
0x600CE0C0     393528       6  Init
```

The following is sample output from the **show memory processor dead**command:

```
Router# show memory processor dead
PC               Total      Count    Name
0x61E4EB70      65588       1  IP Static Rout
0x62332A2C      65588       1  MFI: Clnt SMsg
0x6268DFE4      32820       1  PPP Context Ch
0x62660CCC      32820       1  PPP HANDLE IDs
0x61B9B350      12052       1  IP Addresses
0x614246F8       4148       1  AAA Unique Id Hash Table
0x61BA93CC       3688       1  IPAD DIT chunk
0x63B630A4       2544      12  Autoinstall
0x61824BFC       2084       2  CEF: fib GSB
0x62E82CEC       2052       1  Reg Function 1
0x62E8A028       1824      24  Autoinstall
0x617DE354       1744       2  CEF: paths
0x6149E638       1552       1  String-DB owne
0x6149E490       1552       1  String-DB entr
0x60191180       1216       8  AF entry
0x617EB5AC       1176       2  CEF: pathl
0x62EAE860       1156       1  Event Manager Table
0x6149E4BC        920      12  NameDB String
0x6176BCF4        884       2  Ether OAM subblock
```

The following is sample output from the **show memory processor fragment**command:

```
Router# show memory processor fragment
          Processor memory
Free memory size :  3144348 Number of free blocks:       96
Allocator PC Summary for allocated blocks in pool: Processor
    PC          Total   Count  Name
0x6069A038     262196       1  TACL FLT
0x62224AA8     219188       1  QOS_MODULE_MAIN
0x61648840     131124       1  Init
0x6218DAA4      73780       1  CCSIP_UDP_SOCKET
0x61649288      65588       1  CEF: loadinfo chunk
0x61BFD4B8      65588       1  PPTP mgd timer chunk
0x61EE1050      65588       1  eddri_self_event
0x607C13C4      49204       1  Exec
0x608A0D4C      35208       4  Process Stack
0x6069D804      32052       1  TACL hist
0x61631A90      21444       2  CEF: IPv4 Unicast RPF subblock
0x62BA5DD8      20432       1  Init
0x6086F858      20052       1  RMI-RO_RU Chun
0x608CF99C      20052       1  Managed Chunk Queue Elements
```

The table below describes the significant fields shown in the display.

Table 42: show memory processor fragment Field Descriptions

Field	Description
PC	Program counter.
Total	Total memory allocated by the process (in bytes).
Count	Number of allocations.
Name	Name of the allocating process.

The following is sample output from the **show memory processor free**command:

```
Router# show memory processor free
          Processor memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
             24    Free list 1
66994680 0000000072 66994618 669946FC 000  0        6698FFC8 60699114  Turbo ACr
6698FFC8 0000000072 6698FF60 66990044 000  66994680 659CF6B0 60699114  Turbo ACr
659CF6B0 0000000024 659CF678 659CF6FC 000  6698FFC8 659CF86C 6078A2CC  Init
659CF86C 0000000024 659CF710 659CF8B8 000  659CF6B0 65ADB53C 6078A2CC  Init
65ADB53C 0000000024 65ADB504 65ADB588 000  659CF86C 65ADFC38 6078A2CC  Init
65ADFC38 0000000024 65ADFC00 65ADFC84 000  65ADB53C 65B6C504 6078A2CC  Init
65B6C504 0000000024 65B6C4B8 65B6C550 000  65ADFC38 6593E924 6078A2CC  Init
6593E924 0000000028 6593E8E8 6593E974 000  65B6C504 65CCB054 6078A2CC  Init
65CCB054 0000000024 65CCB01C 65CCB0A0 000  6593E924 65CCBD98 6078A2CC  Init
65CCBD98 0000000028 65CCBD60 65CCBDE8 000  65CCB054 65CCFB70 6078A2CC  Init
65CCFB70 0000000024 65CCFB38 65CCFBBC 000  65CCBD98 65D0BB58 6078A2CC  Init
65D0BB58 0000000024 65D0BB20 65D0BBA4 000  65CCFB70 65D0C5F0 6078A2CC  Init
65D0C5F0 0000000024 65D0C5B8 65D0C63C 000  65D0BB58 65CFF2F4 6078A2CC  Init
65CFF2F4 0000000024 65CFF2BC 65CFF340 000  65D0C5F0 6609B7B8 6078A2CC  Init
6609B7B8 0000000036 6609AFC8 6609B810 000  65CFF2F4 660A0BD4 6078A2CC  Init
```

The table below describes the significant fields shown in the display.

Table 43: show memory processor free Field Descriptions

Field	Description
Address	Hexadecimal address of the block.
Bytes	Size of the block (in bytes).
Prev	Address of the preceding block (should match the address on the preceding row).
Next	Address of the following block (should match the address on the following row).
Ref	Reference count for that memory block, indicating how many different processes are using that block of memory.
PrevF	Address of the preceding free block (if free).
NextF	Address of the following free block (if free).
Alloc PC	Address of the program counter that allocated the block.
what	Name of the process that owns the block.

The following is sample output from the **show memory processor statistics**command:

```
Router# show memory processor statistics
                Head    Total(b)     Used(b)     Free(b)   Lowest(b)  Largest(b)
Processor   6540BBA0   415187836   27216968   387970868   385755044   381633404
      I/O    E000000    33554432    6226336    27328096    27328096    27317852
.
.
.
```

The table below describes the significant fields shown in the display.

Table 44: show memory processor statistics Field Descriptions

Field	Description
Head	Hexadecimal address of the head of the memory allocation chain.
Total(b)	Sum of the used bytes plus free bytes.
Used(b)	Amount of memory in use (in bytes).
Free(b)	Amount of memory not in use (in bytes).
Lowest(b)	Smallest amount of free memory since the last boot (in bytes).
Largest(b)	Size of the largest available free block (in bytes).

show memory scan

To monitor the number and type of parity (memory) errors on your system, use the show memory scan command in EXEC mode.

show memory scan

Syntax Description

This command has no arguments or keywords.

Command Modes

EXEC

Command History

Release	Modification
12.0(4)XE	This command was introduced.
12.0(7)T	This command was implemented in Cisco IOS Release 12.0(7) T.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following example shows a result with no memory errors:

```
Router# show memory scan
Memory scan is on.
No parity error has been detected.
```

If errors are detected in the system, the show memory scan command generates an error report. In the following example, memory scan detected a parity error:

```
Router# show memory scan
Memory scan is on.
Total Parity Errors 1.
Address   BlockPtr   BlckSize  Disposit  Region Timestamp
6115ABCD  60D5D090   9517A4    Scrubed   Local 16:57:09 UTC Thu  Mar 18
```

The table below describes the fields contained in the error report.

Table 45: show memory scan Field Descriptions

Field	Description
Address	The byte address where the error occurred.
BlockPtr	The pointer to the block that contains the error.
BlckSize	The size of the memory block
Disposit	The action taken in response to the error: • BlockInUse--An error was detected in a busy block. • InFieldPrev--An error was detected in the previous field of a block header. • InHeader--An error was detected in a block header. • Linked--A block was linked to a bad list. • MScrubed--The same address was “scrubbed” more than once, and the block was linked to a bad list. • MultiError--Multiple errors have been found in one block. • NoBlkHdr--No block header was found. • NotYet--An error was found; no action has been taken at this time. • Scrubed--An error was “scrubbed.” • SplitLinked--A block was split, and only a small portion was linked to a bad list.
Region	The memory region in which the error was found: • IBSS--image BSS • IData--imagedata • IText--imagetext • local--heap
Timestamp	The time the error occurred.

show memory statistics history

To display the history of memory consumption, use the **show memory statistics history**command in user EXEC or privileged EXEC mode.

show memory statistics history [**table**]

Syntax Description

table	(Optional) Summary of memory consumption history.

Command Modes

User EXEC (>) Privileged EXEC (#)

Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

Examples

The following is sample output from the **show memory statistics history table** command. The field descriptions are self-explanatory.

```
Router# show memory statistics history table
History for Processor memory
Time: 15:48:56.806
Used(b): 422748036 Largest(b): 381064952 Free blocks :291
Maximum memory users for this period
Process Name           Holding   Num Alloc
Virtual Exec             26992          37
TCP Protocols            14460           6
IP Input                  1212           1
Time: 14:42:54.506
Used(b): 422705876 Largest(b): 381064952 Free blocks :296
Maximum memory users for this period
Process Name           Holding   Num Alloc
Exec                  400012740          24
Dead                   1753456          90
Pool Manager            212796         257
Time: 13:37:26.918
Used(b): 20700520 Largest(b): 381064952 Free blocks :196
Maximum memory users for this period
Process Name           Holding   Num Alloc
Exec                      8372           5
Time: 12:39:44.422
Used(b): 20701436 Largest(b): 381064952 Free blocks :193
Time: 11:46:25.135
Used(b): 20701436 Largest(b): 381064952 Free blocks :193
Maximum memory users for this period
Process Name           Holding   Num Alloc
CDP Protocol              3752          25
Time: 10:44:24.342
Used(b): 20701400 Largest(b): 381064952 Free blocks :194
Time: 09:38:53.038
Used(b): 20701400 Largest(b): 381064952 Free blocks :194
Time: 08:33:35.154
Used(b): 20701400 Largest(b): 381064952 Free blocks :194
Time: 07:28:05.987
Used(b): 20701400 Largest(b): 381064952 Free blocks :194
```

```
Time: 06:35:22.878
Used(b): 20701400 Largest(b): 381064952 Free blocks :194
Time: 05:42:14.286
Used(b): 20701400 Largest(b): 381064952 Free blocks :194
Time: 04:41:53.486
Used(b): 20701400 Largest(b): 381064952 Free blocks :194
Time: 03:48:47.891
Used(b): 20701400 Largest(b): 381064952 Free blocks :194
Time: 02:46:32.391
Used(b): 20701400 Largest(b): 381064952 Free blocks :194
Time: 01:54:27.931
Used(b): 20717804 Largest(b): 381064952 Free blocks :189
Time: 01:02:05.535
Used(b): 20717804 Largest(b): 381064952 Free blocks :189
Maximum memory users for this period
Process Name           Holding   Num Alloc
Entity MIB API           67784          16
TTY Background           12928           4
Exec                      7704           3
Time: 00:00:17.936
Used(b): 21011192 Largest(b): 381064952 Free blocks :186
Maximum memory users for this period
Process Name           Holding   Num Alloc
Init                  18653520        6600
CCPROXY_CT              599068          57
Proxy Session Applic    275424          21
History for I/O memory
Time: 15:48:56.809
Used(b):  7455520 Largest(b): 59370080 Free blocks :164
Time: 14:42:54.508
Used(b):  7458064 Largest(b): 59370080 Free blocks :165
Maximum memory users for this period
Process Name           Holding   Num Alloc
Pool Manager            141584         257
Time: 13:37:26.920
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 12:39:44.424
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 11:46:25.137
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 10:44:24.344
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 09:38:53.040
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 08:33:35.156
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 07:28:05.985
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 06:35:22.877
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 05:42:14.285
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 04:41:53.485
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 03:48:47.889
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 02:46:32.389
Used(b):  7297744 Largest(b): 59797664 Free blocks :25
Time: 01:54:27.929
Used(b):  7308336 Largest(b): 59797664 Free blocks :23
Time: 01:02:05.533
Used(b):  7308336 Largest(b): 59797664 Free blocks :23
Time: 00:00:17.937
Used(b):  7308336 Largest(b): 59797664 Free blocks :23
```

```
Maximum memory users for this period
Process Name           Holding   Num Alloc
Init                   7296000         214
Pool Manager               816           3
```

Related Commands

Command	Description
memory statistics history table	Changes the memory log time.

show memory traceback

To display memory traceback information, use the **show memory traceback**command in privileged EXEC mode.

show memory traceback [{*id* | **exclusive** | **totals**}]

Syntax Description

id	(Optional) Traceback ID.
exclusive	(Optional) Displays the memory blocks that have traceback information.
totals	(Optional) Displays information about memory usage of blocks having tracebacks.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.

Usage Guidelines

Before you can enable the **show memory traceback** command, you must configure the **memory record events** command in global configuration mode.

Examples

The following is sample output from the **show memory traceback** command for traceback ID 100:

```
Router# configure terminal

Router(config)# memory record events
Memory event recording already enabled!
Router(config)# exit
Router# show memory traceback 100
Traceback: [100] 0x60630D9Cz  0x60632B50z  0x6063426Cz  0x6063483Cz  0x61AE4910)
```

The following is sample output from the **show memory traceback** command using the **exclusive**keyword:

```
Router# configure terminal

Router(config)# memory record events
Memory event recording already enabled!
Router(config)# exit
Router# show memory traceback exclusive
Address   Size      refcount tid         What
```

```
682E53F4 0005206856 000      T43       (coalesced)
68D2739C 0000002212 000      T85       (coalesced)
```

The table below describes the significant fields shown in the display.

Table 46: show memory traceback Field Descriptions

Field	Description
Address	Hexadecimal address of the block.
Size	Amount of memory, in bytes, used by the task.
refcount	Reference count for the memory block, indicating how many different processes are using that block of memory.
tid	Task ID.
What	Name of the process that owns the block or fragment. Specifies if the block is a fragment or coalesced.

Related Commands

Command	Description
show memory events	Displays recorded memory events.

show memory transient

To display statistics about transient memory, use the **show memory transient**command in user EXEC or privileged EXEC mode.

show memory transient [{**allocating-process** [**totals**] | **dead** [**totals**] | **fragment** [**detail**] | **free** [**totals**] | **statistics** [**history**]}]

Syntax Description

allocating-process	(Optional) Displays allocating memory totals by name.
dead **[totals]**	(Optional) Displays memory totals on dead processes.
fragment **[detail]**	(Optional) Displays memory statistics for fragmented processes.
free **[totals]**	(Optional) Displays statistics on free memory.
statistics **[history]**	(Optional) Displays memory pool history statistics on all processes.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.0	This command was introduced.

Examples

The following is sample output from the **show memory transient**command:

```
Router# show memory transient
          Processor memory
 Address      Bytes     Prev     Next Ref     PrevF    NextF Alloc PC  what
81F99C00 0002236408 00000000 821BBC28 000  829C8104 82776FD0 8060B6D0  (coalesc)
821BBC28 0000020004 81F99C00 821C0A7C 001  -------- -------- 8002D5C0  Managed s
821C0A7C 0000010004 821BBC28 821C31C0 001  -------- -------- 811604C0  List Eles
821C31C0 0000005004 821C0A7C 821C457C 001  -------- -------- 81160500  List Heas
```

The table below describes the significant fields shown in the display.

Table 47: show memory transient Field Descriptions

Field	Description
Address	Hexadecimal address of the block.
Bytes	Size of the block (in bytes).
Prev	Address of the preceding block (should match the address on preceding line).
Next	Address of the following block (should match the address on following line).
Ref	Reference count for that memory block, indicating how many different processes are using that block of memory.
PrevF	Address of the preceding free block (if free).
NextF	Address of the following free block (if free).
Alloc PC	Address of the system call that allocated the block.
what	Name of the process that owns the block, or “(fragment)” if the block is a fragment, or “(coalesced)” if the block was coalesced from adjacent free blocks.

show microcode

To display microcode image information available on line cards, use the **show microcode** command in EXEC mode.

show microcode

Syntax Description

This command has no arguments or keywords.

Command Modes

EXEC

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Examples

The following is sample output from the **show microcode** command:

```
Router# show microcode
Microcode bundled in system
Card    Microcode    Target Hardware    Description
Type    Version      Version
----    ---------    ---------------    -----------
SP         2.3           11.x          SP version 2.3
EIP        1.1            1.x          EIP version 1.1
TRIP       1.2            1.x          TRIP version 1.2
FIP        1.4            2.x          FIP version 1.4
HIP        1.1            1.x          HIP version 1.1
SIP        1.1            1.x          SIP version 1.1
FSIP       1.1            1.x          FSIP version 1.1
```

In the following example for the Cisco 7200 series router, the output from the **show microcode** command lists the hardware types that support microcode download. For each type, the default microcode image name is displayed. If there is a configured default override, that name also is displayed.

```
router# show microcodeMicrocode images for downloadable hardware
HW Type             Microcode image names
-----------------------------------------
ecpa     default    slot0:xcpa26-0
         configured slot0:xcpa26-2
pcpa     default    slot0:xcpa26-4
```

Related Commands

Command	Description
microcode (7000/7500)	Specifies where microcode should be loaded from on Cisco 7500/7000RSP routers.
microcode (7200)	Configures a default override for the microcode that is downloaded to the hardware on a Cisco 7200 series router.

show mls statistics

To display the Multilayer Switching (MLS) statistics for the Internet Protocol (IP), Internetwork Packet Exchange (IPX), multicast, Layer 2 protocol, and quality of service (QoS), use the **show mls statistics** command in user EXEC or privileged EXEC mode.

show mls statistics [**module** *num*]

Syntax Description

module *num*	(Optional) Displays the MLS statistics for a specific module.

Command Default

This command has no default settings.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.2(14)SX	Support for this command was introduced on the Supervisor Engine 720.
12.2(17b)SXA	This command was changed to include the **module** *num* keyword and argument.

Release	Modification
12.2(17d)SXB	Support for this command on the Supervisor Engine 2 was extended to Release 12.2(17d)SXB.
12.2(17d)SXB1	The output was changed to include total packets switched information.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The total packets switched performance displayed is the rate calculated as the average rate in a period within the last 30 seconds.

The ingress ACL denied packet count is displayed in the Total packets L3 Switched field and in the Total packets dropped by ACL field.

The RPF failed packet count is displayed in the Total packets L3 Switched field.

If the IP multicast source sends traffic to any multicast group that does not have an (*,G) entry present in the mroute table, the **show mls statistics** command displays these packets as incrementing in the Total Mcast Packets Switched/Routed field. These packets are dropped in the hardware because there are no receivers for that group and no entry in the mroute table.

Examples

This example shows how to display the MLS statistics for all modules:

```
Router#
show mls statistics
Statistics for Earl in Module 2
L2 Forwarding Engine
  Total packets Switched                : 20273@ 22552 pps
L3 Forwarding Engine
  Total Packets Bridged                 : 20273
  Total Packets FIB Switched            : 7864
  Total Packets ACL Routed              : 0
  Total Packets Netflow Switched        : 0
  Total Mcast Packets Switched/Routed   : 220598
  Total ip packets with TOS changed     : 0
  Total ip packets with COS changed     : 0
  Total non ip packets COS changed      : 0
  Total packets dropped by ACL          : 0
  Total packets dropped by Policing     : 705757744
Statistics for Earl in Module 9
L2 Forwarding Engine
  Total packets Switched                : 16683@ 1 pps
L3 Forwarding Engine
  Total Packets Bridged                 : 0
  Total Packets FIB Switched            : 0
  Total Packets ACL Routed              : 0
  Total Packets Netflow Switched        : 0
  Total Mcast Packets Switched/Routed   : 0
  Total ip packets with TOS changed     : 0
  Total ip packets with COS changed     : 0
  Total non ip packets COS changed      : 0
  Total packets dropped by ACL          : 0
  Total packets dropped by Policing     : 277949053
Router#
```

This example shows how to display the MLS statistics for a specific module:

```
Router#
show mls statistics module 1
Statistics for Earl in Module 1
L2 Forwarding Engine
  Total packets Switched                : 2748166@ 22332 pps
>>
L3 Forwarding Engine
  Total Packets Bridged                 : 92750@ 34 pps
  Total Packets FIB Switched            : 7
  Total Packets ACL Routed              : 0
  Total Packets Netflow Switched        : 0
  Total Mcast Packets Switched/Routed   : 3079200
  Total ip packets with TOS changed     : 0
  Total ip packets with COS changed     : 0
  Total non ip packets COS changed      : 0
  Total packets dropped by ACL          : 0
  Total packets dropped by Policing     : 0
  Total Unicast RPF failed packets      : 0
Errors
  MAC/IP length inconsistencies         : 0
  Short IP packets received             : 0
  IP header checksum errors             : 0
  MAC/IPX length inconsistencies        : 0
  Short IPX packets received            : 0
Router
#
```

Related Commands

Command	Description
show mls asic	display the application-specific integrated circuit (ASIC) version
show mls df-table	Displays information about the DF table.
show mls ip	Displays the Multilayer Switching (MLS) IP information.
show mls ipx	Displays the Multilayer Switching (MLS) IPX information.
show mls qos	Displays Multilayer Switching (MLS) quality of service (QoS) information
show mls statistics	Displays the Multilayer Switching (MLS) statistics for the Internet Protocol (IP)

show module

To display the module status and information, use the **show** module command in user EXEC or privileged EXEC mode.

show module [{**mod-num** | **all** | **provision** | **version**}]

Syntax Description

mod -num	(Optional) Number of the module.
all	(Optional) Displays the information for all modules.
provision	(Optional) Displays the status about the module provisioning.
version	(Optional) Displays the version information.

Command Default

This command has no default settings.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.2(14)SX	Support for this command was introduced on the Supervisor Engine 720.
12.2(17d)SXB	Support for this command on the Supervisor Engine 2 was extended to Release 12.2(17d)SXB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

In the Mod Sub-Module fields, the **show module** command displays the supervisor engine number but appends the uplink daughter card's module type and information.

Entering the **show module** command with no arguments is the same as entering the **show module all** command.

Examples

This example shows how to display information for all modules on a Cisco 7600 series router that is configured with a Supervisor Engine 720:

```
Router#
show module

Mod Ports Card Type Model Serial No.
--- ----- ------------------------------------- ------------------ -----------
1 48 CEF720 48 port 10/100/1000mb Ethernet WS-X6748-GE-TX SAL0843557C
2 48 48-port 10/100/1000 RJ45 EtherModule WS-X6148A-GE-45AF SAL1109HZW9
3 48 48-port 10/100/1000 RJ45 EtherModule WS-X6148A-GE-45AF SAL1114KYZ7
4 48 48 port 10/100 mb RJ45 WS-X6348-RJ-45 SAL0543DGZ1
6 2 Supervisor Engine 720 (Active) WS-SUP720-3B SAL1016KASS
7 48 48-port 10/100 mb RJ45 WS-X6148-45AF SAL08321X1H
8 4 CEF720 4 port 10-Gigabit Ethernet WS-X6704-10GE SAL08528ADQ
9 48 48-port 100FX SFP Ethernet Module WS-X6148-FE-SFP SAD090208MB
Mod MAC addresses Hw Fw Sw Status
--- ---------------------------------- ------ ------------ ------------ -------
1 0012.005c.86e0 to 0012.005c.870f 2.1 12.2(14r)S5 12.2(33)SXH Ok
2 001b.0ce4.9fb0 to 001b.0ce4.9fdf 2.2 8.4(1) 8.7(0.22)SXH Ok
3 001b.534f.0540 to 001b.534f.056f 2.2 8.4(1) 8.7(0.22)SXH Ok
4 0007.4f6c.69f8 to 0007.4f6c.6a27 5.0 5.4(2) 8.7(0.22)SXH Ok
6 0017.9441.44cc to 0017.9441.44cf 5.2 8.4(2) 12.2(33)SXH Ok
7 0011.bb0e.c260 to 0011.bb0e.c28f 1.1 5.4(2) 8.7(0.22)SXH Ok
8 0012.da89.a43c to 0012.da89.a43f 2.0 12.2(14r)S5 12.2(33)SXH Ok
9 0030.f273.baf0 to 0030.f273.bb1f 3.0 8.4(1) 8.7(0.22)SXH Ok
Mod Sub-Module Model Serial Hw Status
---- --------------------------- ------------------ ----------- ------- -------
1 Centralized Forwarding Card WS-F6700-CFC SAL08363HL6 2.0 Ok
2 IEEE Voice Daughter Card WS-F6K-48-AF SAL1108HRB1 2.3 Ok
3 IEEE Voice Daughter Card WS-F6K-48-AF SAL1114KV3P 2.3 Ok
4 Inline Power Module WS-F6K-VPWR 1.0 Ok
6 Policy Feature Card 3 WS-F6K-PFC3B SAL1015K00Q 2.3 Ok
6 MSFC3 Daughterboard WS-SUP720 SAL1016KBY3 2.5 Ok
7 IEEE Voice Daughter Card WS-F6K-FE48-AF SAL08311GGL 1.1 Ok
8 Centralized Forwarding Card WS-F6700-CFC SAL0902040K 2.0 Ok
Mod Online Diag Status
---- -------------------
1 Bypass
2 Bypass
3 Bypass
4 Bypass
```

```
6 Bypass
7 Bypass
8 Bypass
9 Bypass
Router#
```

This example shows how to display information for a specific module:

```
Router#
show module 2
Mod Ports Card Type                              Model              Serial No.
--- ----- ---------------------------------------- ------------------ -----------
  5     2 Supervisor Engine 720 (Active)         WS-SUP720-BASE     SAD0644030K
Mod MAC addresses                       Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
  5 00e0.aabb.cc00 to 00e0.aabb.cc3f   1.0    12.2(2003012 12.2(2003012 Ok
Mod Sub-Module                  Model           Serial           Hw     Status
--- --------------------------- --------------- --------------- ------- -------
  5 Policy Feature Card 3       WS-F6K-PFC3     SAD0644031P     0.302   Ok
  5 MSFC3 Daughtercard          WS-SUP720       SAD06460172     0.701
Mod Online Diag Status
--- -------------------
  5 Not Available
Router#
```

This example shows how to display version information:

```
Router#
show module version

Mod Port Model              Serial #    Versions
--- ---- ------------------ ----------- -------------------------------------
  2 0    WS-X6182-2PA                   Hw : 1.0
                     Fw : 12.2(20030125:231135)
                     Sw : 12.2(20030125:231135)
  4 16   WS-X6816-GBIC      SAD04400CEE Hw : 0.205
         WS-F6K-DFC3A       SAD0641029Y Hw : 0.501
                     Fw : 12.2(20020828:202911)
                     Sw : 12.2(20030125:231135)
  6 2    WS-X6K-SUP3-BASE   SAD064300GU Hw : 0.705
                     Fw : 7.1(0.12-Eng-02)TAM
                     Sw : 12.2(20030125:231135)
                     Sw1: 8.1(0.45)KIS
         WS-X6K-SUP3-PFC3   SAD064200VR Hw : 0.701
                     Fw : 12.2(20021016:001154)
                     Sw : 12.2(20030125:231135)
         WS-F6K-PFC3        SAD064300M7 Hw : 0.301
  9 48   WS-X6548-RJ-45     SAD04490BAC Hw : 0.301
                     Fw : 6.3(1)
                     Sw : 7.5(0.30)CFW11
Router#
```

This example shows how to display module provisioning information:

```
Router# show module provision
Module Provision
  1    dynamic
  2    dynamic
  3    dynamic
  4    dynamic
  5    dynamic
  6    dynamic
  7    dynamic
```

```
  8    dynamic
  9    dynamic
 10    dynamic
 11    dynamic
 12    dynamic
 13    dynamic
Router#
```

Related Commands

Command	Description
show interfaces	Displays the status and statistics for the interfaces in the chassis.
show environment alarm	Displays the information about the environmental alarm.
show fm summary	Displays a summary of FM Information.
show environment status	Displays the information about the operational FRU status.

show monitor event-trace

To display event trace messages for Cisco IOS software subsystem components, use the **show monitor event-trace** command in privileged EXEC mode.

show monitor event-trace [**all-traces**] *component* {**all** | **back** *hour:minute* | **clock** *hour:minute* | **from-boot** *seconds* | **latest** | **parameters**}

Syntax Description

all-traces	(Optional) Displays all event trace messages in memory to the console.
component	(Optional) Name of the Cisco IOS software subsystem component that is the object of the event trace. To get a list of components that support event tracing in this release, use the **monitor event-trace ?** command.
all	Displays all event trace messages currently in memory.
back *mmm* | *hhh:mm*}	Specifies how far back from the current time you want to view messages. For example, you can gather messages from the last 30 minutes. The time argument is specified either in minutes or in hours and minutes format (mmm or hh:mm).
clock *hh:mm*	Displays event trace messages starting from a specific clock time in hours and minutes format (hh:mm).
date	(Optional) Day of the month.
month	(Optional) Displays the month of the year.
from-boot *seconds*	Displays event trace messages starting from a specified number of seconds after booting (uptime). To display the uptime, in seconds, enter the **show monitor event-trace***component***from-boot ?** command.
latest	Displays only the event trace messages since the last **show monitor event-trace** command was entered.

component	(Optional) Name of the Cisco IOS software subsystem component that is the object of the event trace. To get a list of components that support event tracing in this release, use the **monitor event-trace ?** command.
parameters	Displays the trace parameters. The only parameter displayed is the size (number of trace messages) of the trace file.
detail	(Optional) Displays detailed trace information.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.0(18)S	This command was introduced.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S. The **show monitor event-trace cef** comand replaced the **show cef events**and **show ip cef events** commands.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE. The **spa** component keyword was added to support online insertion and removal (OIR) event messages for shared port adapters (SPAs). The **bfd** keyword was added for the *component* argument to display trace messages relating to the Bidirectional Forwarding Detection (BFD) feature.
12.4(4)T	Support for the **bfd** keyword was added for Cisco IOS Release 12.4(4)T.
12.0(31)S	Support for the **bfd** keyword was added for Cisco IOS Release 12.0(31)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB and implemented on the Cisco 10000 series routers.
12.4(9)T	The **cfd** keyword was added as an entry for the *component* argument to display trace messages relating to crypto fault detection.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.
IOS XE Fuji 16.9.1	The subscriber ppp component was added.

Usage Guidelines

Use the **show monitor event-trace** command to display trace message information.

The trace function is not locked while information is being displayed to the console, which means that new trace messages can accumulate in memory. If entries accumulate faster than they can be displayed, some messages can be lost. If this happens, the **show monitor event-trace** command will generate a message

indicating that some messages might be lost; however, messages will continue to display on the console. If the number of lost messages is excessive, the **show monitor event-trace**command will stop displaying messages.

Use the **bfd** keyword for the *component* argument to display trace messages relating to the BFD feature.

Use the **cfd** keyword for the *component* argument to display trace messages relating to the crypto fault detection feature. This keyword displays the contents of the error trace buffers in an encryption data path.

Examples

IPC Component Example

The following is sample output from the **show monitor event-trace***component* command for the interprocess communication (IPC) component. Notice that each trace message is numbered and is followed by a time stamp (derived from the device uptime). Following the time stamp is the component-specific message data.

```
Router# show monitor event-trace ipc

3667:  6840.016:Message type:3 Data=0123456789
3668:  6840.016:Message type:4 Data=0123456789
3669:  6841.016:Message type:5 Data=0123456789
3670:  6841.016:Message type:6 Data=0123456
```

BFD Component for Cisco IOS Release 12.2(18)SXE, 12.0(31)S, and 12.4(4)T

Use the **show monitor event-trace bfd all**command to display logged messages for important BFD events in the recent past. The following trace messages show BFD session state changes:

```
Router# show monitor event-trace bfd all
    3d03h: EVENT: Session [172.16.10.2,172.16.10.1,Fa6/0,1], event Session
           create, state Unknown -> Fail
    3d03h: EVENT: Session [172.16.10.2,172.16.10.1,Fa6/0,1], state Fail -> Down
            (from LC)
    3d03h: EVENT: Session [172.16.10.2,172.16.10.1,Fa6/0,1], state Down -> Init
            (from LC)
    3d03h: EVENT: Session [172.16.10.2,172.16.10.1,Fa6/0,1], state Init -> Up
           (from LC)
    3d07h: EVENT: Session [172.16.10.2,172.16.10.1,Fa6/0,2], event Session
           create, state Unknown -> Fail
    3d07h: EVENT: Session [172.16.10.2,172.16.10.1,Fa6/0,2], state Fail -> Down
            (from LC)
    3d07h: EVENT: Session [172.16.10.2,172.16.10.1,Fa6/0,2], state Down -> Up
           (from LC)
```

To display trace information for all components configured for event tracing on the networking device, enter the **show monitor event-trace all-traces** command. In this example, separate output is provided for each event, and message numbers are interleaved between the events.

```
Router# show monitor event-trace all-traces

Test1 event trace:
3667: 6840.016:Message type:3 Data=0123456789
3669: 6841.016:Message type:4 Data=0123456789
3671: 6842.016:Message type:5 Data=0123456789
3673: 6843.016:Message type:6 Data=0123456789
```

```
Test2 event trace:
3668: 6840.016:Message type:3 Data=0123456789
3670: 6841.016:Message type:4 Data=0123456789
3672: 6842.016:Message type:5 Data=0123456789
3674: 6843.016:Message type:6 Data=0123456789
```

SPA Component Example

The following is sample output from the **show monitor event-trace** *component* **latest** command for the **spa** component:

```
Router# show monitor event-trace spa latest
00:01:15.364: subslot 2/3: 4xOC3 POS SPA, TSM Event:inserted  New state:wait_psm
_ready
     spa type 0x440
00:02:02.308: subslot 2/0: not present, TSM Event:empty  New state:remove
     spa type 0x0, fail code 0x0(none)
00:02:02.308: subslot 2/0: not present, TSM Event:remove_complete  New state:idle
00:02:02.308: subslot 2/1: not present, TSM Event:empty  New state:remove
     spa type 0x0, fail code 0x0(none)
00:02:02.308: subslot 2/1: not present, TSM Event:remove_complete  New state:idle
00:02:02.308: subslot 2/2: not present, TSM Event:empty  New state:remove
     spa type 0x0, fail code 0x0(none)
00:02:02.308: subslot 2/2: not present, TSM Event:remove_complete  New state:idle
00:02:02.312: subslot 2/3: not present(plugin 4xOC3 POS SPA), TSM Event:empty  New
state:remove
     spa type 0x0, fail code 0x0(none)
00:02:02.312: subslot 2/3: not present, TSM Event:remove_complete  New state:idle
```

Cisco Express Forwarding Component Examples

If you select Cisco Express Forwarding as the component for which to display event messages, you can use the following additional arguments and keywords: **show monitor event-trace cef** [**events** | **interface** | **ipv6** | **ipv4**][**all**].

The following example shows the IPv6 or IPv4 events related to the Cisco Express Forwarding component. Each trace message is numbered and is followed by a time stamp (derived from the device uptime). Following the time stamp is the component-specific message data.

```
Router# show monitor event-trace cef ipv6 all
00:00:24.612:  [Default] *::*/*'00             New FIB table          [OK]
Router# show monitor event-trace cef ipv4 all
00:00:24.244:  [Default] 127.0.0.81/32'01       FIB insert             [OK]
```

In the following example, all event trace messages for the Cisco Express Forwarding component are displayed:

```
Router# show monitor event-trace cef events all
00:00:18.884: SubSys  fib_ios_chain init
00:00:18.884: Inst    unknown -> RP
00:00:24.584: SubSys  fib init
00:00:24.592: SubSys  fib_ios init
00:00:24.592: SubSys  fib_ios_if init
00:00:24.596: SubSys  ipv4fib init
00:00:24.608: SubSys  ipv4fib_ios init
00:00:24.612: SubSys  ipv6fib_ios init
00:00:24.620: Flag    IPv4 CEF enabled set to yes
```

```
00:00:24.620: Flag    0x7BF6B62C set to yes
00:00:24.620: Flag    IPv4 CEF switching enabled set to yes
00:00:24.624: GState  CEF enabled
00:00:24.628: SubSys  ipv4fib_les init
00:00:24.628: SubSys  ipv4fib_pas init
00:00:24.632: SubSys  ipv4fib_util init
00:00:25.304: Process Background created
00:00:25.304: Flag    IPv4 CEF running set to yes
00:00:25.304: Process Background event loop enter
00:00:25.308: Flag    IPv4 CEF switching running set to yes
```

The following example shows Cisco Express Forwarding interface events:

```
Router# show monitor event-trace cef interface all
00:00:24.624: <empty>      (sw  4) Create   new
00:00:24.624: <empty>      (sw  4) SWIDBLnk FastEthernet0/0(4)
00:00:24.624: Fa0/0        (sw  4) NameSet
00:00:24.624: <empty>      (hw  1) Create   new
00:00:24.624: <empty>      (hw  1) HWIDBLnk FastEthernet0/0(1)
00:00:24.624: Fa0/0        (hw  1) NameSet
00:00:24.624: <empty>      (sw  3) Create   new
00:00:24.624: <empty>      (sw  3) SWIDBLnk FastEthernet0/1(3)
00:00:24.624: Fa0/1        (sw  3) NameSet
00:00:24.624: <empty>      (hw  2) Create   new
```

Cisco Express Forwarding Component Examples for Cisco 10000 Series Routers Only

The following example shows the IPv4 events related to the Cisco Express Forwarding component. Each trace message is numbered and is followed by a time stamp (derived from the device uptime). Following the time stamp is the component-specific message data.

```
Router# show monitor event-trace cef ipv4 all
00:00:48.244:  [Default] 127.0.0.81/32'01      FIB insert               [OK]
```

In the following example, all event trace message for the Cisco Express Forwarding component are displayed:

```
Router# show monitor event-trace cef events all
00:00:18.884: SubSys  fib_ios_chain init
00:00:18.884: Inst    unknown -> RP
00:00:24.584: SubSys  fib init
00:00:24.592: SubSys  fib_ios init
00:00:24.592: SubSys  fib_ios_if init
00:00:24.596: SubSys  ipv4fib init
00:00:24.608: SubSys  ipv4fib_ios init
00:00:24.620: Flag    IPv4 CEF enabled set to yes
00:00:24.620: Flag    0x7BF6B62C set to yes
00:00:24.620: Flag    IPv4 CEF switching enabled set to yes
00:00:24.624: GState  CEF enabled
00:00:24.628: SubSys  ipv4fib_les init
00:00:24.628: SubSys  ipv4fib_pas init
00:00:24.632: SubSys  ipv4fib_util init
00:00:25.304: Process Background created
00:00:25.304: Flag    IPv4 CEF running set to yes
00:00:25.304: Process Background event loop enter
00:00:25.308: Flag    IPv4 CEF switching running set to yes
```

The following examples show Cisco Express Forwarding interface events:

```
Router# show monitor event-trace cef interface all

00:00:24.624: <empty>      (sw  4) Create   new
00:00:24.624: <empty>      (sw  4) SWIDBLnk FastEthernet1/0/0(4)
00:00:24.624: Fa0/0        (sw  4) NameSet
00:00:24.624: <empty>      (hw  1) Create   new
00:00:24.624: <empty>      (hw  1) HWIDBLnk FastEthernet1/0/0(1)
00:00:24.624: Fa0/0        (hw  1) NameSet
00:00:24.624: <empty>      (sw  3) Create   new
00:00:24.624: <empty>      (sw  3) SWIDBLnk FastEthernet1/1/0(3)
00:00:24.624: Fa0/1        (sw  3) NameSet
00:00:24.624: <empty>      (hw  2) Create   new
```

CFD Component for Cisco IOS Release 12.4(9)T

To troubleshoot errors in an encryption datapath, enter the **show monitor event-trace cfd all command. In this example, events are shown separately, each beginning with a time stamp, followed by data from the error trace buffer. Cisco Technical Assistence Center (TAC) engineers can use this information to diagnose the cause of the errors.**

Note If no packets have been dropped, this command does not display any output.

```
Router# show monitor event-trace cfd all
00:00:42.452: 450000B4 00060000 FF33B306 02020203 02020204 32040000 F672999C
        00000001 7A7690C2 A0A4F8BC E732985C D6FFDCC8 00000001 C0902BD0
        A99127AE 8EAA22D4
00:00:44.452: 450000B4 00070000 FF33B305 02020203 02020204 32040000 F672999C
        00000002 93C01218 2325B697 3C384CF1 D6FFDCC8 00000002 BFA13E8A
        D21053ED 0F62AB0E
00:00:46.452: 450000B4 00080000 FF33B304 02020203 02020204 32040000 F672999C
        00000003 7D2E11B7 A0BA4110 CC62F91E D6FFDCC8 00000003 7236B930
        3240CA8C 9EBB44FF
00:00:48.452: 450000B4 00090000 FF33B303 02020203 02020204 32040000 F672999C
        00000004 FB6C80D9 1AADF938 CDE57ABA D6FFDCC8 00000004 E10D8028
        6BBD748F 87F5E253
00:00:50.452: 450000B4 000A0000 FF33B302 02020203 02020204 32040000 F672999C
        00000005 697C8D9D 35A8799A 2A67E97B D6FFDCC8 00000005 BC21669D
        98B29FFF F32670F6
00:00:52.452: 450000B4 000B0000 FF33B301 02020203 02020204 32040000 F672999C
        00000006 CA18CBC4 0F387FE0 9095C27C D6FFDCC8 00000006 87A54811
        AE3A0517 F8AC4E64
```

Related Commands

Command	Description
monitor event-trace (EXEC)	Controls event trace functions for a specified Cisco IOS software subsystem component.
monitor event-trace (global)	Configures event tracing for a specified Cisco IOS software subsystem component.
monitor event-trace dump-traces	Saves trace messages for all event traces currently enabled on the networking device.

show monitor event-trace flexvpn

To display event trace messages for FlexVPN, use the **show monitor event-trace flexvpn** command in privileged EXEC mode.

show monitor event-trace flexvpn {**all** | **back** *hour:minute* | **clock** *hour:minute* | **crypto** | **from-boot** *seconds* | **latest** | **merged** | **nhrp** | **tunnel**}

Syntax Description

all	Show all the traces in the current buffer.
back { *mmm* | *hh:mm*}	Specifies how far back from the current time you want to view messages. The time argument is specified either in minutes or in hours and minutes format (mmm or hh:mm). For example, you can gather messages from the last 30 minutes.
clock *hh:mm*	Displays event trace messages starting from a specific clock time in hours and minutes format (hh:mm).
crypto	Displays all crypto events, such as IKEv2, IPsec, and PKI.
from-boot *seconds*	Displays event trace messages starting from a specified number of seconds after boot (uptime).
latest	Displays only the event trace messages since the last **show monitor event-trace flexvpn** command was executed.
merged	Show entries in all event traces sorted by time.
nhrp	Show the next-hop resolution protocol (NHRP) trace.
tunnel	Display all tunnel events.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
IOS XE Gibraltar 16.11.1	This command was introduced.

Usage Guidelines

Use the **show monitor event-trace flexvpn** command to display trace message information related to FlexVPN.

The following example shows command output when using the **all** option.

```
Router#show monitor event-trace flexvpn all

crypto_socket:

*Jul  8 06:01:35.582: CRYPTO-TP-EVENT:CRYPTO-SS  Virtual-Access1: local address : 192.0.2.1
 remote address : 192.0.2.10 socket is UP
*Jul  8 06:01:59.991: CRYPTO-TP-EVENT:CRYPTO-SS  Virtual-Access1: local address : 192.0.2.1
 remote address : 192.0.2.10 socket is DOWN
```

```
*Jul  8 06:02:00.781: CRYPTO-TP-EVENT:CRYPTO-SS  Virtual-Access1: local address : 192.0.2.1
 remote address : 192.0.2.10 socket is UP

nhrp_event:

nhrp_error:

nhrp_exception:

flexvpn_client:

*Jul  8 23:02:04.917: FlexVPN (flex):Current Event: EV_RESET, State Change: IDLE -> ST_NO_CHG,
 Next Event: EV_NO_EVENT
*Jul  8 23:02:04.935: FlexVPN (flex):Current Event: EV_CHECK_CFG, State Change: IDLE ->
ST_NO_CHG, Next Event: EV_NO_EVENT
*Jul  8 23:02:05.063: FlexVPN (flex):Current Event: EV_CHECK_CFG, State Change: IDLE ->
ST_NO_CHG, Next Event: EV_NO_EVENT

ikev2_event:

*Jul  8 06:01:29.793: SA ID:1 SESSION ID:1 Remote: 192.0.2.10/500 Local: 192.0.2.1/500  (R)
 Received IKEv2 IKE_SA_INIT Exchange REQUEST
*Jul  8 06:01:29.793: SA ID:1 SESSION ID:1 Remote: 192.0.2.10/500 Local: 192.0.2.1/500
(R) MsgID = 0 CurState: IDLE CurEvent: EV_RECV_I     NIT RetVal: RC_FALSE NextState: R_INIT
 NextEvent: EV_VERIFY_MSG
*Jul  8 06:01:29.795: SA ID:1 SESSION ID:1 Remote: 192.0.2.10/500 Local: 192.0.2.1/500  (R)
 Sending IKEv2 IKE_SA_INIT Exchange RESPONSE

ikev2_error:

ikev2_exception:


ipsec_event:

*Jul  8 06:01:35.308: IPSEC-EVENT:IPSEC-SEND-KMI:  Session ID : 2, KMI Sent:
KEY_MGR_IKMP_READY, KMI source: Crypto IKEv2, KMI dest: IPSEC key engine
*Jul  8 06:01:35.308: IPSEC-EVENT:IPSEC-RECV-KMI:  Session ID : 2, KMI Received:
KEY_MGR_IKMP_READY, KMI source: Crypto IKEv2, KMI dest: IPSEC key engine, loc:
                         192.0.2.1, rem: 192.0.2.10, port loc/rem: 500/500
*Jul  8 06:01:35.363: IPSEC-EVENT:IPSEC-SEND-KMI:  Session ID : 2, KMI Sent:
KEY_ENG_IPSEC_READY, KMI source: IPSEC key engine, KMI dest: Crypto IKEv2

ipsec_error:


ipsec_exception:


pki_event:

*Jul  8 05:59:22.382: EST client initialized.
*Jul  8 05:59:42.481: EST client process started.
*Jul  8 06:01:29.979: Trustpoint- ID:validation status - CRYPTO_VALID_CERT_WITH_WARNING
*Jul  8 06:01:59.982: Trustpoint- ID:validation status - CRYPTO_VALID_CERT_WITH_WARNING
pki_error:
*Jul  8 05:59:42.481: PKI timers have not been initialized due to non-authoritative system
 clock. Ensure system clock is configured/updated.
```

Related Commands

Command	Description
monitor event-trace (EXEC)	Controls event trace functions for a specified Cisco IOS software subsystem component.
monitor event-trace (global)	Configures event tracing for a specified Cisco IOS software subsystem component.
monitor event-trace dump-traces	Saves trace messages for all event traces currently enabled on the networking device.