

Process Health Monitoring

This chapter describes how to manage and monitor the health of various components of your router. It contains the following sections:

- Monitoring Control Plane Resources, on page 1
- Monitoring Hardware Using Alarms, on page 6

Monitoring Control Plane Resources

The following sections explain the details of memory and CPU monitoring from the perspective of the Cisco IOS process and the overall control plane:

- Avoiding Problems Through Regular Monitoring, on page 1
- Cisco IOS Process Resources, on page 1
- Overall Control Plane Resources, on page 5

Avoiding Problems Through Regular Monitoring

Processes should provide monitoring and notification of their status/health to ensure correct operation. When a process fails, a syslog error message is displayed and either the process is restarted or the router is rebooted. A syslog error message is displayed when a monitor detects that a process is stuck or has crashed. If the process can be restarted, it is restarted; else, the router is restarted.

Monitoring system resources enables you to detect potential problems before they occur, thus avoiding outages. It also establishes a baseline for a normal system load. You can use this information as a basis for comparison, when you upgrade hardware or software to see if the upgrade has affected resource usage.

Cisco IOS Process Resources

You can view CPU utilization statistics on active processes and see the amount of memory being used in these processes using the **show memory** command and the **show process cpu** command. These commands provide a representation of memory and CPU utilization from the perspective of only the Cisco IOS process; they do not include information for resources on the entire platform. When the **show memory** command is used in a system with 4 GB RAM running a single Cisco IOS process, the following memory usage is displayed:

```
Router# show memory
Tracekey : 1#33e0077971693714bd2b0bc347d77489
Address Bytes Prev Next Ref PrevF NextF what Alloc PC

Head Total(b) Used(b) Free(b) Lowest(b) Largest(b)
Processor 7F68ECD010 728952276 281540188 447412088 445683380 234766720
lsmpi_io 7F6852A1A8 6295128 6294304 824 824 412
Dynamic heap limit(MB) 200 Use(MB) 0



Processor memory

Address Bytes Prev Next Ref PrevF NextF what Alloc PC
7F68ECD010 0000000568 00000000 7F68ECD2A0 001 -------- -------- *Init* :400000+60E37C4
7F68ECD2A0 0000032776 7F68ECD010 7F68ED5300 001 -------- -------- Managed Chunk Q
:400000+60D12A8
7F68ED5300 0000000056 7F68ECD2A0 7F68ED5390 001 -------- -------- *Init* :400000+3B0C610
7F68ED5390 0000012808 7F68ED5300 7F68ED85F0 001 -------- -------- *Init* :400000+B8A5D64
Address Bytes Prev Next Ref PrevF NextF what Alloc PC
7F68ED85F0 0000032776 7F68ED5390 7F68EE0650 001 -------- -------- List Elements
:400000+60A4A9C
7F68EE0650 0000032776 7F68ED85F0 7F68EE86B0 001 -------- -------- List Headers :400000+60A4AD8
7F68EE86B0 0000032776 7F68EE0650 7F68EF0710 001 -------- -------- IOSXE Process S
:400000+11924CC
7F68EF0710 0000032776 7F68EE86B0 7F68EF8770 001 -------- -------- IOSXE Queue Pro
:400000+1192510
7F68EF8770 0000065544 7F68EF0710 7F68F087D0 001 -------- -------- IOSXE Queue Bal
:400000+1192554
7F68F087D0 0000000328 7F68EF8770 7F68F08970 001 -------- -------- *Init* :400000+B89E1D8
7F68F08970 0000000328 7F68F087D0 7F68F08B10 001 -------- -------- *Init* :400000+B89E1D8
7F68F08B10 0000000328 7F68F08970 7F68F08CB0 001 -------- -------- *Init* :400000+B89E1D8
7F68F08CB0 0000000360 7F68F08B10 7F68F08E70 001 -------- -------- Process Events
:400000+60F9CD4
7F68F08E70 0000000056 7F68F08CB0 7F68F08F00 001 -------- -------- SDB String :400000+605981C
7F68F08F00 0000000080 7F68F08E70 7F68F08FA8 001 -------- -------- Init :400000+60599E4
Address Bytes Prev Next Ref PrevF NextF what Alloc PC
7F68F08FA8 0000036872 7F68F08F00 7F68F12008 001 -------- -------- *Init* :400000+11891E8
7F68F12008 0000010008 7F68F08FA8 7F68F14778 001 -------- -------- Platform VM Pag
:400000+11AD244
7F68F14778 0000002008 7F68F12008 7F68F14FA8 001 -------- -------- *Init*
iosd_crb_ir1101_unix:7F8EB59000+5CC1C
7F68F14FA8 0000200712 7F68F14778 7F68F46008 001 -------- -------- Interrupt Stack
:400000+11891E8
7F68F46008 0000003008 7F68F14FA8 7F68F46C20 001 -------- -------- Watched Semapho
:400000+60FE448
7F68F46C20 0000000328 7F68F46008 7F68F46DC0 001 -------- -------- *Init* :400000+B89E1D8
7F68F46DC0 0000000096 7F68F46C20 7F68F46E78 001 -------- -------- Init :400000+60599E4
7F68F46E78 0000000216 7F68F46DC0 7F68F46FA8 001 -------- -------- *Init* :400000+60ED228
7F68F46FA8 0000036872 7F68F46E78 7F68F50008 001 -------- -------- *Init* :400000+11891E8
7F68F50008 0000000896 7F68F46FA8 7F68F503E0 001 -------- -------- Watched Message
:400000+60FE4A8
7F68F503E0 0000002008 7F68F50008 7F68F50C10 001 -------- -------- Watcher Message
:400000+60FE4D8
Address Bytes Prev Next Ref PrevF NextF what Alloc PC
7F68F50C10 0000000360 7F68F503E0 7F68F50DD0 001 -------- -------- Process Events
:400000+60F9CD4
7F68F50DD0 0000000184 7F68F50C10 7F68F50EE0 001 -------- -------- *Init* :400000+60ED918
7F68F50EE0 0000000112 7F68F50DD0 7F68F50FA8 001 -------- -------- *Init* :400000+60B57CC
7F68F50FA8 0000036872 7F68F50EE0 7F68F5A008 001 -------- -------- *Init* :400000+11891E8
7F68F5A008 0000002336 7F68F50FA8 7F68F5A980 001 -------- -------- Process Array
:400000+6102A4C
7F68F5A980 0000000184 7F68F5A008 7F68F5AA90 001 -------- -------- *Init* :400000+60ED918
```

```
7F68F5AA90 0000000184 7F68F5A980 7F68F5ABA0 001 -------- -------- *Init* :400000+60ED918
7F68F5ABA0 0000000184 7F68F5AA90 7F68F5ACB0 001 -------- -------- *Init* :400000+60ED918
7F68F5ACB0 0000000184 7F68F5ABA0 7F68F5ADC0 001 -------- -------- *Init* :400000+60ED918
7F68F5ADC0 0000000184 7F68F5ACB0 7F68F5AED0 001 -------- -------- *Init* :400000+60ED918
```

The **show process cpu** command displays Cisco IOS CPU utilization average:

```
Router# show process cpu
CPU utilization for five seconds: 0%/0%; one minute: 0%; five minutes: 0%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
1 0 17 0 0.00% 0.00% 0.00% 0 Chunk Manager
2 552 1205 458 0.00% 0.00% 0.00% 0 Load Meter
3 0 1 0 0.00% 0.00% 0.00% 0 PKI Trustpool
4 0 1 0 0.00% 0.00% 0.00% 0 Retransmission o
5 0 1 0 0.00% 0.00% 0.00% 0 IPC ISSU Dispatc
6 36 13 2769 0.00% 0.00% 0.00% 0 RF Slave Main Th
7 0 1 0 0.00% 0.00% 0.00% 0 EDDRI_MAIN
8 0 1 0 0.00% 0.00% 0.00% 0 RO Notify Timers
9 4052 920 4404 0.23% 0.09% 0.06% 0 Check heaps
10 12 101 118 0.00% 0.00% 0.00% 0 Pool Manager
11 0 1 0 0.00% 0.00% 0.00% 0 DiscardQ Backgro
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
12 0 2 0 0.00% 0.00% 0.00% 0 Timers
13 0 163 0 0.00% 0.00% 0.00% 0 WATCH_AFS
14 0 2 0 0.00% 0.00% 0.00% 0 ATM AutoVC Perio
15 0 2 0 0.00% 0.00% 0.00% 0 ATM VC Auto Crea
16 76 3024 25 0.00% 0.00% 0.00% 0 IOSXE heartbeat
17 0 13 0 0.00% 0.00% 0.00% 0 DB Lock Manager
18 0 1 0 0.00% 0.00% 0.00% 0 DB Notification
19 0 1 0 0.00% 0.00% 0.00% 0 IPC Apps Task
20 0 1 0 0.00% 0.00% 0.00% 0 ifIndex Receive
21 36 1210 29 0.00% 0.00% 0.00% 0 IPC Event Notifi
22 72 5904 12 0.00% 0.00% 0.00% 0 IPC Mcast Pendin
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
23 0 1 0 0.00% 0.00% 0.00% 0 Platform appsess
24 0 101 0 0.00% 0.00% 0.00% 0 IPC Dynamic Cach
25 16 1210 13 0.00% 0.00% 0.00% 0 IPC Service NonC
26 0 1 0 0.00% 0.00% 0.00% 0 IPC Zone Manager
27 64 5904 10 0.00% 0.00% 0.00% 0 IPC Periodic Tim
28 76 5904 12 0.00% 0.00% 0.00% 0 IPC Deferred Por
29 0 1 0 0.00% 0.00% 0.00% 0 IPC Process leve
30 0 1 0 0.00% 0.00% 0.00% 0 IPC Seat Manager
31 8 346 23 0.00% 0.00% 0.00% 0 IPC Check Queue
32 0 1 0 0.00% 0.00% 0.00% 0 IPC Seat RX Cont
33 0 1 0 0.00% 0.00% 0.00% 0 IPC Seat TX Cont
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
34 48 606 79 0.00% 0.00% 0.00% 0 IPC Keep Alive M
35 28 1210 23 0.00% 0.00% 0.00% 0 IPC Loadometer
36 0 1 0 0.00% 0.00% 0.00% 0 IPC Session Deta
37 0 1 0 0.00% 0.00% 0.00% 0 SENSOR-MGR event
38 4 606 6 0.00% 0.00% 0.00% 0 Compute SRP rate
39 0 1 0 0.00% 0.00% 0.00% 0 MEMLEAK PROCESS
40 0 1 0 0.00% 0.00% 0.00% 0 ARP Input
41 112 6331 17 0.00% 0.00% 0.00% 0 ARP Background
42 0 2 0 0.00% 0.00% 0.00% 0 ATM Idle Timer
43 0 1 0 0.00% 0.00% 0.00% 0 ATM ASYNC PROC
44 0 1 0 0.00% 0.00% 0.00% 0 CEF MIB API
--More--
...
show process cpu platform sorted

CPU utilization for five seconds: 11%, one minute: 12%, five minutes: 12%
```

```
Core 0: CPU utilization for five seconds: 1%, one minute: 3%, five minutes: 3%
Core 1: CPU utilization for five seconds: 1%, one minute: 3%, five minutes: 3%
Core 2: CPU utilization for five seconds: 1%, one minute: 1%, five minutes: 1%
Core 3: CPU utilization for five seconds: 42%, one minute: 42%, five minutes: 42%
Pid PPid 5Sec 1Min 5Min Status Size Name
------------------------------------------------------------------------------
18246 17700 34% 34% 34% S 272500 qfp-ucode-sparr
18297 16477 1% 1% 1% S 165768 fman_fp_image
9992 9121 1% 1% 1% S 743608 linux_iosd-imag
27122 26048 0% 0% 0% S 8460 nginx
26048 25864 0% 0% 0% S 19252 nginx
25928 1 0% 0% 0% S 2960 rotee
25864 1 0% 0% 0% S 3532 pman.sh
24212 2 0% 0% 0% S 0 kworker/u8:0
19648 8282 0% 0% 0% S 220 sleep
19635 10903 0% 0% 0% S 212 sleep
18121 17675 0% 0% 0% S 10968 ngiolite
17979 1 0% 0% 0% S 1660 rotee
17863 2 0% 0% 0% S 0 kworker/1:0
17859 1 0% 0% 0% S 2836 rotee
17737 17095 0% 0% 0% S 56828 iomd
17700 13380 0% 0% 0% S 3556 pman.sh
17675 12798 0% 0% 0% S 3524 pman.sh
17518 16854 0% 0% 0% S 15024 hman
17312 1 0% 0% 0% S 2828 rotee
17095 12798 0% 0% 0% S 3568 pman.sh
17085 1 0% 0% 0% S 2876 rotee
16942 2 0% 0% 0% S 0 kworker/0:1
16892 14768 0% 0% 0% S 108952 cpp_cp_svr
16854 13380 0% 0% 0% S 3568 pman.sh
16716 1 0% 0% 0% S 2996 rotee
16664 15963 0% 0% 0% S 51096 cpp_sp_svr
16477 13380 0% 0% 0% S 3540 pman.sh
16326 15536 0% 0% 0% S 39852 cpp_ha_top_leve
16270 1 0% 0% 0% S 2972 rotee
15963 13380 0% 0% 0% S 3528 pman.sh
15779 15163 0% 0% 0% S 55208 cpp_driver
15730 1 0% 0% 0% S 1640 rotee
15536 13380 0% 0% 0% S 3528 pman.sh
15412 1 0% 0% 0% S 1716 rotee
15274 14681 0% 0% 0% S 15004 hman
15163 13380 0% 0% 0% S 3624 pman.sh
15083 14361 0% 0% 0% S 26792 cman_fp
15057 1 0% 0% 0% S 1660 rotee
14891 1 0% 0% 0% S 2868 rotee
14768 13380 0% 0% 0% S 3568 pman.sh
14722 14127 0% 0% 0% S 27536 cmcc
14717 14108 0% 0% 0% S 15220 btman
14681 12798 0% 0% 0% S 3572 pman.sh
14627 1 0% 0% 0% S 2996 rotee
14361 13380 0% 0% 0% S 3596 pman.sh
14338 1 0% 0% 0% S 2984 rotee
14314 1 0% 0% 0% S 2824 rotee
14155 13577 0% 0% 0% S 15128 btman
14127 12798 0% 0% 0% S 3612 pman.sh
14108 13380 0% 0% 0% S 3572 pman.sh
13813 13380 0% 0% 0% S 252 inotifywait
--More--
```

Overall Control Plane Resources

Control plane memory and CPU utilization on each control processor allows you to keep a tab on the overall control plane resources. You can use the **show platform software status control-processor brief** command (summary view) or the **show platform software status control-processor** command (detailed view) to view control plane memory and CPU utilization information.

All control processors should show status, Healthy. Other possible status values are Warning and Critical. Warning indicates that the router is operational, but that the operating level should be reviewed. Critical implies that the router is nearing failure.

If you see a Warning or Critical status, take the following actions:

- Reduce the static and dynamic loads on the system by reducing the number of elements in the configuration or by limiting the capacity for dynamic services.
- Reduce the number of routes and adjacencies, limit the number of ACLs and other rules, reduce the number of VLANs, and so on.

The following sections describe the fields in the **show platform software status control-processor** command output.

Load Average

Load average represents the process queue or process contention for CPU resources. For example, on a single-core processor, an instantaneous load of 7 would mean that seven processes are ready to run, one of which is currently running. On a dual-core processor, a load of 7 would mean that seven processes are ready to run, two of which are currently running.

Memory Utilization

Memory utilization is represented by the following fields:

- Total—Total system memory
- Used—Consumed memory
- Free—Available memory
- Committed—Virtual memory committed to processes

CPU Utilization

CPU utilization is an indication of the percentage of time the CPU is busy, and is represented by the following fields:

- CPU—Allocated processor
- User—Non-Linux kernel processes
- System—Linux kernel process
- Nice—Low-priority processes
- Idle—Percentage of time the CPU was inactive
- IRQ—Interrupts

- SIRQ—System Interrupts
- IOwait—Percentage of time CPU was waiting for I/O

Example: show platform software status control-processor Command

The following are some examples of using the **show platform software status control-processor** command:

```
Router# show platform software status control-processor
RP0: online, statistics updated 4 seconds ago
Load Average: healthy
1-Min: 0.29, status: healthy, under 5.00
5-Min: 0.51, status: healthy, under 5.00
15-Min: 0.54, status: healthy, under 5.00
Memory (kb): healthy
Total: 4038072
Used: 2872136 (71%), status: healthy
Free: 1165936 (29%)
Committed: 2347228 (58%), under 90%
Per-core Statistics
CPU0: CPU Utilization (percentage of time spent)
User: 1.00, System: 0.70, Nice: 0.00, Idle: 97.88
IRQ: 0.30, SIRQ: 0.10, IOwait: 0.00
CPU1: CPU Utilization (percentage of time spent)
User: 0.70, System: 0.30, Nice: 0.00, Idle: 98.48
IRQ: 0.30, SIRQ: 0.20, IOwait: 0.00
CPU2: CPU Utilization (percentage of time spent)
User: 0.20, System: 1.11, Nice: 0.00, Idle: 98.27
IRQ: 0.40, SIRQ: 0.00, IOwait: 0.00
CPU3: CPU Utilization (percentage of time spent)
User: 8.23, System: 24.37, Nice: 0.00, Idle: 58.00
IRQ: 9.26, SIRQ: 0.11, IOwait: 0.00
```

```
Router# show platform software status control-processor brief
Load Average
Slot Status 1-Min 5-Min 15-Min
RP0 Healthy 0.28 0.46 0.52

Memory (kB)
Slot Status Total Used (Pct) Free (Pct) Committed (Pct)
RP0 Healthy 4038072 2872672 (71%) 1165400 (29%) 2349820 (58%)

CPU Utilization
Slot CPU User System Nice Idle IRQ SIRQ IOwait
RP0 0 0.70 0.20 0.00 98.58 0.30 0.20 0.00
1 1.10 0.90 0.00 97.59 0.30 0.10 0.00
2 0.40 1.31 0.00 97.87 0.40 0.00 0.00
3 8.00 26.55 0.00 56.33 8.99 0.11 0.00
```

Monitoring Hardware Using Alarms

This section contains the following:

Router Design and Monitoring Hardware

The router sends alarm notifications when problems are detected, allowing you to monitor the network remotely. You do not need to use **show** commands to poll devices on a routine basis; however, you can perform onsite monitoring if you choose.

BootFlash Disk Monitoring

The bootflash disk must have enough free space to store two core dumps. This condition is monitored, and if the bootflash disk is too small to store two core dumps, a syslog alarm is generated, as shown in the following example:

```
Oct  6 14:10:56.292: %FLASH_CHECK-3-DISK_QUOTA: R0/0: flash_check: Flash disk quota exceeded

[free space is 1429020 kB] - Please clean up files on bootflash.
```

Approaches for Monitoring Hardware Alarms

This section contains the following:

Viewing the Console or Syslog for Alarm Messages

The network administrator can monitor alarm messages by reviewing alarm messages sent to the system console or to a system message log (syslog).

Enabling the logging alarm Command

The **logging** alarm command must be enabled for the system to send alarm messages to a logging device, such as the console or a syslog. This command is not enabled by default.

You can specify the severity level of the alarms to be logged. All the alarms at and above the specified threshold generate alarm messages. For example, the following command sends only critical alarm messages to logging devices:

```
Router(config)# logging alarm critical
```

If alarm severity is not specified, alarm messages for all severity levels are sent to logging devices.

Report Alarms Through SNMP

SNMP is an application-layer protocol that provides a standardized framework and a common language used for monitoring and managing devices in a network.

SNMP provides notification of faults, alarms, and conditions that might affect services. It allows a network administrator to access router information through a network management system (NMS) instead of reviewing logs, polling devices, or reviewing log reports.

To use SNMP to get alarm notification, use the following MIBs:

- ENTITY-MIB, RFC4133 (required for the CISCO-ENTITY-ALARM-MIB, ENTITY-STATE-MIB and CISCO-ENTITY-SENSOR-MIB to work)
- CISCO-ENTITY-ALARM-MIB
- ENTITY-STATE-MIB

- CISCO-ENTITY-SENSOR-MIB (for transceiver environmental alarm information, which is not provided through the CISCO-ENTITY-ALARM-MIB)

Yang Support for IO Ports

This feature increases the compatibility between the Command Line Interface and the Yang Model. Cisco IOS-XE Yang Data Models are found here:

https://github.com/YangModels/yang/tree/master/vendor/cisco/xe

Each release has a directory. The two modules for Digital IO are Cisco-IOS-XE-digital-io-oper and Cisco-IOS-XE-digitalio.

The following are relevant IOS-XE CLI commands available:

Show Commands

- show run
- show alarm
- show led

Configuration Commands

- alarm contact attach-to-iox
- no alarm contact attach-to-iox
- alarm contact 1 enable enable
- no alarm contact <1-4> enable
- alarm contact <1-4> application <wet | dry>
- no alarm contact <1-4> application
- alarm contact <1-4> description <alarm description>
- no alarm contact <1-4> description
- alarm contact <1-4> severity <critical | major | minor | none>
- no alarm contact <1-4> severity
- alarm contact <1-4> threshold <1600-2700>
- no alarm contact <1-4> threshold
- alarm contact <1-4> trigger <closed | open>
- no alarm contact <1-4> trigger
- alarm contact <1-4> output <1 | 0>
- alarm contact <1-4> output relay temperature <critical | major | minor>
- alarm contact <1-4> output relay input-alarm <0-4>

- no alarm contact <1-4> output

SNMP MIB for Digital I/O

Digital I/O is similar to the ALARM IN and ALARM OUT supported in other IR devices. On other devices, ALARM IN is a dedicated input and the ALARM OUT is a dedicated output. With Digital I/O it can be input or output. There are 4 Digital I/O available on the IR1001 with an IRM-1100 Expansion Module.

MIB support will reflect the show alarm output for digital I/O only.

CISCO-DIGITAL-IO-MIB.my will have 4 digital I/O nodes. Each digital I/O node will have corresponding attributes like description, enable, severity, application, output, threshold, trigger leaf nodes for each digital I/O nodes.

SNMP MIB supports the show power CLI

SNMP MIB support for the **show power** CLI is available through a new mib file: CISCO-ENTITY-SENSOR-MIB.my

The following is an example of the **show power** CLI:

```
#show power
Main PSU :
    Total Power Consumed: 8.77 Watts
    Configured Mode : N/A
    Current runtime state same : N/A
    PowerSupplySource : External PS
```

The following is an example of the CISCO-ENTITY-SENSOR-MIB.my MIB:

```
SensorDataType (INTEGER) watts(6)
SensorDataScale (INTEGER) milli(8)
SensorValue(INTEGER) 8770
```

Use the following commands to configure:

```
Router#config term
Router#(config) snmp-server community public RW
Router#(config) end
```