

Collect CPU Utilization on Cisco IOS Devices with SNMP

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

[Procedure for Devices with Single CPU](#)

[Example](#)

[Procedure for Devices with Multiple CPUs](#)

[Example](#)

[Related Information](#)

Introduction

This document describes how to collect CPU utilization on Cisco IOS® devices that use Simple Network Management Protocol (SNMP).

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is only for devices that run Cisco IOS software.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Conventions

Refer to [Cisco Technical Tips Conventions](#) for more information on document conventions.

Procedure for Devices with Single CPU

Critical router functions, like routing protocol processing and process packet switching, are handled in memory and share the CPU. Thus, if CPU utilization is very high, it is possible that a routing update cannot be handled, or a process-switching packet is dropped. From the [CISCO-PROCESS-MIB](#), the [cpmCPUTotal5minRev](#) MIB object value reports the percentage of the processor in use over a five-minute average.



Note: Use the [Cisco MIB Locator](#) to find information about the MIB on a particular platform or software release.

The [cpmCPUTotal5minRev](#) MIB object provides a more accurate view of the performance of the router over time than the MIB objects [cpmCPUTotal1minRev](#) and [cpmCPUTotal5secRev](#). These MIB objects are not accurate because they look at CPU at one minute and five second intervals, respectively. These MIBs enable you to monitor the trends and plan the capacity of your network. The recommended baseline rising threshold for [cpmCPUTotal5minRev](#) is 90 percent. Based on the platform, some routers that run at 90 percent, can exhibit performance degradation versus a high-end router, which can operate fine.

- **cpmCPUTotal5secRev** (.1.3.6.1.4.1.9.9.109.1.1.1.1.6): The overall CPU busy percentage in the last five-second period. This object deprecates the object [cpmCPUTotal5sec](#) and increases the value range to (0..100).
- **cpmCPUTotal1minRev** (.1.3.6.1.4.1.9.9.109.1.1.1.1.7): The overall CPU busy percentage in the last one-minute period. This object deprecates the object [cpmCPUTotal1min](#) and increases the value range to (0..100).
- **cpmCPUTotal5minRev** (.1.3.6.1.4.1.9.9.109.1.1.1.1.8): The overall CPU busy percentage in the last five-minute period. This object deprecates the object [cpmCPUTotal5min](#) and increases the value

range to (0..100).

This table shows the new MIBs and their objects beside the old MIBs and objects they replace:

Version	Cisco IOS Software releases 12.2(3.5) or later	Cisco IOS Software releases later to 12.0(3)T and prior to 12.2(3.5)
MIB	CISCO-PROCESS-MIB	CISCO-PROCESS-MIB
Objects	cpmCPUTotal5minRev (.1.3.6.1.4.1.9.9.109.1.1.1.1.8) cpmCPUTotal1minRev (.1.3.6.1.4.1.9.9.109.1.1.1.1.7) cpmCPUTotal5secRev (.1.3.6.1.4.1.9.9.109.1.1.1.1.6)	cpmCPUTotal5min (.1.3.6.1.4.1.9.9.109.1.1.1.1.5) cpmCPUTotal1min (.1.3.6.1.4.1.9.9.109.1.1.1.1.4) cpmCPUTotal5sec (.1.3.6.1.4.1.9.9.109.1.1.1.1.3)

Example

Here is a typical output from the **show processes CPU** command on a router that runs Cisco IOS Software Release 12.0(9):

```
<#root>

Router#

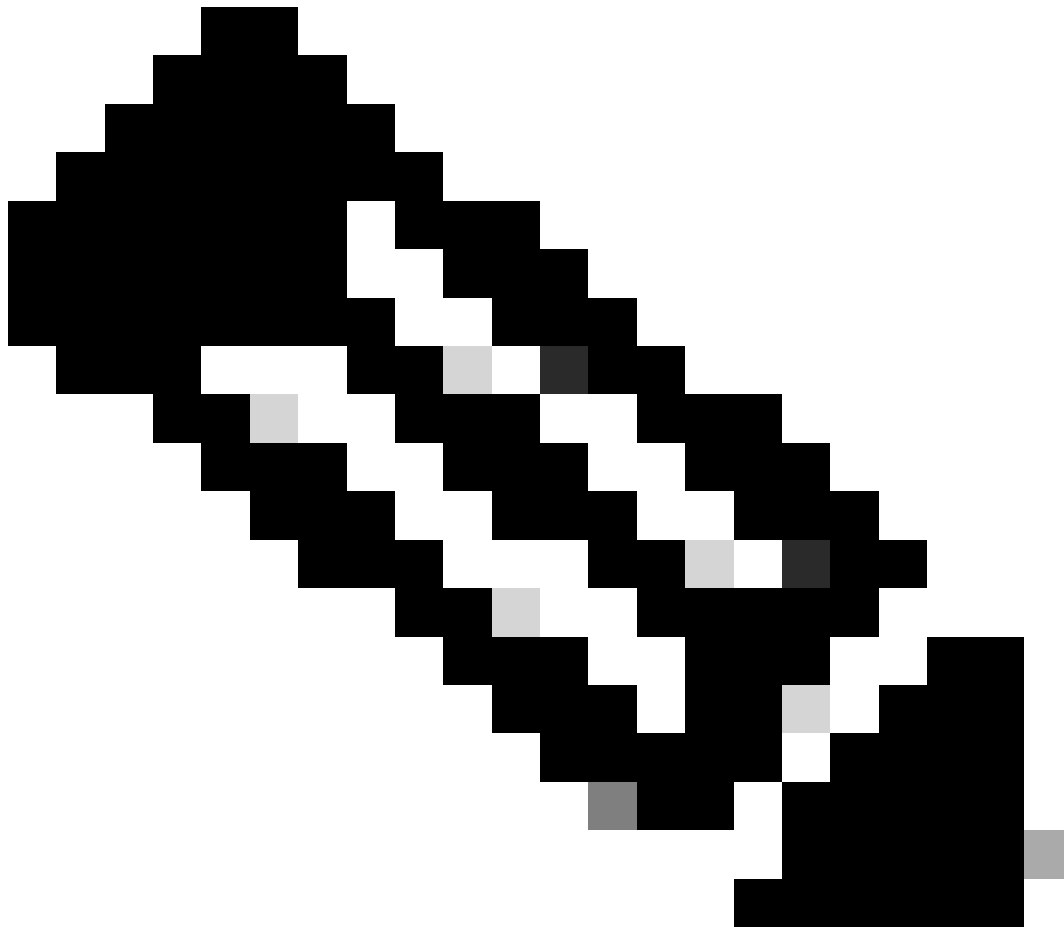
show processes CPU

CPU utilization for five seconds: 2%/1%; one minute: 1%; five minutes: 1%

PID  Runtime(ms)  Invoked  uSecs   5Sec   1Min   5Min  TTY  Process
  1      164      137902    1    0.00%  0.00%  0.00%  0  Load Meter
  2      100       119     840    0.57%  0.11%  0.02%  2  Virtual Exec
  3   468644     81652   5739    0.00%  0.04%  0.05%  0  Check heaps
  4         0         1         0    0.00%  0.00%  0.00%  0  Pool Manager
  5         0         2         0    0.00%  0.00%  0.00%  0  Timers
  6         0         2         0    0.00%  0.00%  0.00%  0  Serial Background
  7         0         1         0    0.00%  0.00%  0.00%  0  OIR Handler
  8         0         1         0    0.00%  0.00%  0.00%  0  IPC Zone Manager
  9      348     689225    0    0.00%  0.00%  0.00%  0  IPC Periodic Tim
 10         0         1         0    0.00%  0.00%  0.00%  0  IPC Seat Manager
 11   175300    332916   526    0.00%  0.02%  0.00%  0  ARP Input
 12     3824    138903   27    0.00%  0.00%  0.00%  0  HC Counter Timer
 13         0         2         0    0.00%  0.00%  0.00%  0  DDR Timers
 14         0         1         0    0.00%  0.00%  0.00%  0  Entity MIB API
 15         0         1         0    0.00%  0.00%  0.00%  0  SERIAL A'detect
 16         0         1         0    0.00%  0.00%  0.00%  0  Microcode Loader
 17         0         1         0    0.00%  0.00%  0.00%  0  IP Crashinfo Inp
--<snip>--
```

From the output, this is the relevant information:

 **Note:** On the basis of the Cisco IOS Software Release that runs on the device, use appropriate MIB objects.



Note: Only registered Cisco users can access internal Cisco tools and information.

- The CPU utilization over the last five seconds [also available through the object [busyPer](#) (.1.3.6.1.4.1.9.2.1.56)]

```
%snmpwalk -v2c -c public 172.16.99.1 .1.3.6.1.4.1.9.2.1.56  
  
!--- SNMP Query  
  
enterprises.9.2.1.56.0 = 2  
  
!--- Response
```

- The percentage of CPU time at interrupt level (fast-switched packets), over a five-second period. If you take the difference between the first and second, you arrive at the five-second percentage the router spends at the process level. In this case, the router spends one percent at the process level over the last five seconds (process- switched packets - No MIB variable).
- The CPU utilization over the last minute [also available through the object [avgBusy1\(.1.3.6.1.4.1.9.2.1.57\)](#)]

```
%snmpwalk -v2c -c public 172.16.99.1 .1.3.6.1.4.1.9.2.1.57
```

```
!--- SNMP Query
```

```
enterprises.9.2.1.57.0 = 3
```

```
!--- Response
```

- The CPU utilization over the last five minutes [also available through the object [avgBusy5\(.1.3.6.1.4.1.9.2.1.58\)](#)]

```
%snmpwalk -v2c -c public 172.16.99.1 .1.3.6.1.4.1.9.2.1.58
```

```
!--- SNMP Query
```

```
enterprises.9.2.1.58.0 = 1
```

```
!--- Response
```

When you poll CPU utilization variables and any other SNMP variables, the actual CPU utilization is affected. Sometimes, the utilization is 99 percent when you continuously poll the variable at one-second intervals. It is an overkill to poll so frequently but take into consideration the impact to CPU when you determine how frequently you want to poll the variable.

Procedure for Devices with Multiple CPUs

If your Cisco IOS device has several CPUs, you must use [CISCO-PROCESS-MIB](#) and its object [cpmCPUTotal5minRev](#) from the table called [cpmCPUTotalTable](#), indexed with [cpmCPUTotalIndex](#). This table allows [CISCO-PROCESS-MIB](#) to keep CPU statistics for different physical entities in the router, like different CPU chips, group of CPUs, or CPUs in different modules/cards. In case of a single CPU, [cpmCPUTotalTable](#) has only one entry.

Information about different physical entities in the router is stored in the [entPhysicalTable](#) of RFC 2737 standard-based [ENTITY-MIB](#). You can link between two tables ([cpmCPUTotalTable](#) and [entPhysicalTable](#)) easily: each row of [cpmCPUTotalTable](#) has an object [cpmCPUTotalPhysicalIndex](#) that keeps the value of the [entPhysicalIndex](#) (index of [entPhysicalTable](#)), and points to the entry in [entPhysicalTable](#), correspondent to the physical entity for which these CPU statistics are maintained.

This implies that the Cisco IOS device must support both [CISCO-PROCESS-MIB](#) and [ENTITY-MIB](#) for you

to be able to retrieve relevant information about CPU utilization. The only case where you do not need to have or use [ENTITY-MIB](#) is when you only have a single CPU.

Example

Monitor the use of multiple CPUs in the chassis (RSP and two VIPs). The same applies to GSR linecards.

1. Poll [cpmCPUTotal5min](#) (.1.3.6.1.4.1.9.9.109.1.1.1.1.5) to get "overall CPU busy percentage in the last 5 minute period" for all CPUs in the chassis. The output shows that the device has three CPUs, utilized for 10%, 1%, and 2% during last 5 minutes.

```
%snmpwalk -v2c -c public 172.16.0.1 .1.3.6.1.4.1.9.9.109.1.1.1.1.5
```

```
!--- SNMP Query
```

```
enterprises.9.9.109.1.1.1.1.5.1 = 10  
enterprises.9.9.109.1.1.1.1.5.8 = 1  
enterprises.9.9.109.1.1.1.1.5.9 = 2
```

```
!--- Response
```



Note: On the basis of the Cisco IOS Software Release that runs on the device, use appropriate MIB objects.

2. In order to identify the physical entity to which these values correspond, poll [cpmCPUTotalPhysicalIndex](#) (.1.3.6.1.4.1.9.9.109.1.1.1.1.2). You see three physical entities with indexes 9, 25 and 28:

```
%snmpwalk -v2c -c public 172.16.0.1 .1.3.6.1.4.1.9.9.109.1.1.1.1.2
```

```
!--- SNMP Query
```

```
enterprises.9.9.109.1.1.1.1.2.1 = 9  
enterprises.9.9.109.1.1.1.1.2.8 = 25  
enterprises.9.9.109.1.1.1.1.2.9 = 28
```

```
!--- Response
```

3. In order to identify the particular card to which each physical entry is related, poll correspondent [entPhysicalName](#) (.1.3.6.1.2.1.47.1.1.1.1.7) entry, with the exact indexes 9, 25, 28 from Step 2, as a last digit. You see that RSP is utilized for 10%, and VIPs in slots 4 and 6 are utilized for one and two percent.

```
%snmpwalk -v2c -c public 172.16.0.1 .1.3.6.1.2.1.47.1.1.1.1.7.9
```

```
!--- SNMP Query
```

47.1.1.1.1.7.9 = "RSP at Slot 2"

!--- Response

```
%snmpwalk -v2c -c public 172.16.0.1 .1.3.6.1.2.1.47.1.1.1.1.7.25
```

!--- SNMP Query

47.1.1.1.1.7.25 = "Line Card 4"

!--- Reponse

```
%snmpwalk -v2c -c public 172.16.0.1 .1.3.6.1.2.1.47.1.1.1.1.7.28
```

!--- SNMP Query

47.1.1.1.1.7.28 = "Line Card 6"

!--- Response

Related Information

- [Understanding VIP CPU Running at 99% and Rx-Side Buffering](#)
- [Troubleshooting High CPU Utilization on Cisco Routers](#)
- [SNMP Community String Indexing](#)
- [Simple Network Management Protocol Index](#)
- [Cisco Technical Support & Downloads](#)