

# Comment collecter les informations d'utilisation du CPU sur les périphériques de Cisco IOS à l'aide de SNMP

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Conventions](#)

[Procédure pour des périphériques avec la CPU simple](#)

[Exemple](#)

[Procédure pour des périphériques avec de plusieurs CPU](#)

[Exemple](#)

[Procédure pour la CPU des cartes de RUELLE de Catalyst 5000](#)

[Informations connexes](#)

## [Introduction](#)

Ce document décrit comment recueillir l'utilisation du processeur sur les périphériques de Cisco IOS® qui utilisent le protocole de gestion de réseau simple (SNMP).

## [Conditions préalables](#)

### [Conditions requises](#)

Aucune spécification déterminée n'est requise pour ce document.

### [Composants utilisés](#)

Les informations dans ce document sont seulement pour les périphériques qui exécutent le logiciel de Cisco IOS. L'exemple c7500 avec de plusieurs CPU utilise la version du logiciel Cisco IOS 12.0(22)S3.

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

### [Conventions](#)

Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous à [Conventions relatives aux conseils techniques Cisco](#).

## Procédure pour des périphériques avec la CPU simple

Des fonctions routeur essentielles, comme le traitement de protocole de routage et la commutation de paquets de processus, sont manipulées dans la mémoire et partagent la CPU. Ainsi, si l'utilisation du processeur est très élevée, il est possible qu'une mise à jour de routage ne puisse pas être manipulée ou un paquet de processus-commutation est lâché. [Du CISCO-PROCESS-MIB](#), la valeur d'objet MIB [cpmCPUTotal5minRev](#) signale le pourcentage du processeur en service au-dessus d'une moyenne de cinq-minute.

L'objet MIB [cpmCPUTotal5minRev](#) fournit une vue plus précise de la représentation du routeur au fil du temps que les objets [cpmCPUTotal1minRev](#) et [cpmCPUTotal5secRev](#) MIB. Ces objets MIB ne sont pas précis parce qu'ils regardent la CPU à une minute et aux cinq seconde intervalles, respectivement. Ce MIB te permet de surveiller les tendances et de prévoir la capacité de votre réseau. Le seuil montant recommandé de spécification de base pour [cpmCPUTotal5minRev](#) est de 90 pour cent. Selon la plate-forme, quelques Routeurs qui fonctionnent à 90 pour cent, par exemple, 2500s, peuvent montrer la dégradation de représentation contre un routeur haut de gamme, par exemple, la gamme 7500, qui peut fonctionner bien.

- [cpmCPUTotal5secRev](#) (.1.3.6.1.4.1.9.9.109.1.1.1.1.6) : Le pourcentage occupé global CPU pendant la dernière période de cinq secondes. Cet objet désapprouve l'objet [cpmCPUTotal5sec](#) et augmente la plage de valeur à (0..100).
- [cpmCPUTotal1minRev](#) (.1.3.6.1.4.1.9.9.109.1.1.1.1.7) : Le pourcentage occupé global CPU pendant la dernière période d'one-minute. Cet objet désapprouve l'objet [cpmCPUTotal1min](#) et augmente la plage de valeur à (0..100).
- [cpmCPUTotal5minRev](#) (.1.3.6.1.4.1.9.9.109.1.1.1.1.8) : Le pourcentage occupé global CPU pendant la dernière période de cinq-minute. Cet objet désapprouve l'objet [cpmCPUTotal5min](#) et augmente la plage de valeur à (0..100).

Cette table affiche le nouveau MIB et leurs objets près du vieux MIB et objets qu'ils remplacent :

<b>Version</b>	Versions de logiciels 12.2(3.5) ou ultérieures de Cisco IOS	Versions logicielles de Cisco IOS plus tard à 12.0(3)T et avant 12.2(3.5)	Versions logicielles de Cisco IOS avant 12.0(3)T
<b>MIB</b>	<a href="#">CISCO-PROCESS-MIB</a>	<a href="#">CISCO-PROCESS-MIB</a>	<a href="#">OLD-CISCO-CPU-MIB</a>
<b>Objets</b>	<a href="#">cpmCPUTotal5minRev</a>	<a href="#">cpmCPUTotal5min</a> (.1.3.6.1.4.1.9.9.109.1.1.1.1.8)	<a href="#">avgBusy5</a> (.1.3.6.1.4.1.9.9.109.1.1.1.1.9)

<a href="#">al5min Rev</a> (.1.3.6 .1.4.1. 9.9.10 9.1.1. 1.1.8)	1.1.1.1.5)	.9.2.1.58)
<a href="#">cpmCPUTotal1min Rev</a> (.1.3.6 .1.4.1. 9.9.10 9.1.1. 1.1.7)	<a href="#">cpmCPUTotal1min</a> (.1.3.6.1.4.1.9.9.109. 1.1.1.1.4)	<a href="#">avgBusy1</a> (.1.3.6.1.4.1 .9.2.1.57)
<a href="#">cpmCPUTotal5sec Rev</a> (.1.3.6 .1.4.1. 9.9.10 9.1.1. 1.1.6)	<a href="#">cpmCPUTotal5sec</a> (.1.3.6.1.4.1.9.9.109. 1.1.1.1.3)	<a href="#">busyPer</a> (.1.3.6.1.4.1 .9.2.1.56)

## Exemple

Voici un résultat typique de la commande de **show processes cpu** sur un routeur qui exécute le Logiciel Cisco IOS version 12.0(9) :

```
Router# show processes CPU
```

**CPU utilization for five seconds: 2%A/1%B; one minute: 1%C; five minutes: 1%D**

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>--
```

De la sortie, c'est les informations pertinentes :

**Note:** Sur la base de la version du logiciel Cisco IOS qui fonctionne sur le périphérique, utilisez les objets appropriés MIB.

- L'utilisation du processeur au cours des cinq dernières secondes [aussi disponibles par le [busyPer d'objet \(.1.3.6.1.4.1.9.2.1.56\)](#)]

```
Router# show processes CPU
```

```
CPU utilization for five seconds: 2%A/1%B; one minute: 1%C; five minutes: 1%D
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>--
```

- Le pourcentage du temps- CPU au niveau de priorité d'interruption (paquets à commutation rapide), sur une période de cinq secondes. Si vous prenez la différence entre le premier et le deuxième, vous arrivez au cinq-deuxième pourcentage que le routeur passe au niveau de processus. Dans ce cas, le routeur passe un pour cent au niveau de processus au cours des cinq dernières secondes (paquets commutés par processus - aucune variable MIB).
- L'utilisation du processeur au-dessus du de dernière minute [aussi disponible par l'objet [avgBusy1 \(.1.3.6.1.4.1.9.2.1.57\)](#)]

```
Router# show processes CPU
```

```
CPU utilization for five seconds: 2%A/1%B; one minute: 1%C; five minutes: 1%D
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>--
```

- L'utilisation du processeur au cours des cinq dernières minutes [aussi disponibles par l'objet [avgBusy5 \(.1.3.6.1.4.1.9.2.1.58\)](#)]

```
Router# show processes CPU
```

CPU utilization for five seconds: 2%A/1%B; one minute: 1%C; five minutes: 1%D

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>--

Quand vous votez des variables d'utilisation du processeur et toutes les autres variables SNMP, l'utilisation du processeur réelle est affectée. Parfois, l'utilisation est de 99 pour cent quand vous votez continuellement la variable aux un-deuxièmes intervalles. C'est une sursaturation d'armes à voter tellement fréquemment, mais prend en compte l'incidence à la CPU quand vous déterminez comment fréquemment vous voulez voter la variable.

## [Procédure pour des périphériques avec de plusieurs CPU](#)

Si votre périphérique IOS a plusieurs CPU, vous devez utiliser [CISCO-PROCESS-MIB](#) et son objet [cpmCPUTotal5minRev de la](#) table appelée [cpmCPUTotalTable](#), répertorié avec le [cpmCPUTotalIndex](#). Cette table permet à [CISCO-PROCESS-MIB](#) pour maintenir des statistiques CPU pour différentes entités physiques dans le routeur, comme différentes puces CPU, groupe de CPU, ou CPU dans différents modules/cartes. En cas de CPU simple, [cpmCPUTotalTable](#) a seulement une entrée.

Des informations sur différentes entités physiques dans le routeur sont stockées dans l'[entPhysicalTable de la](#) conformité aux normes [ENTITY-MIB RFC 2737](#). Vous pouvez joindre entre deux tables ([cpmCPUTotalTable](#) et [entPhysicalTable](#)) facilement : chaque ligne de [cpmCPUTotalTable](#) a un [cpmCPUTotalPhysicalIndex d'](#)objet qui garde la valeur de l'[entPhysicalIndex](#) (index d'[entPhysicalTable](#)), et des points à l'entrée dans [entPhysicalTable](#), correspondant à l'entité physique pour laquelle ces des statistiques CPU sont mises à jour.

Ceci implique que le périphérique IOS doit prendre en charge [CISCO-PROCESS-MIB](#) et [ENTITY-MIB](#) pour que vous puissiez récupérer les informations pertinentes au sujet de l'utilisation du processeur. Le seul cas où vous n'avez pas besoin d'avoir ou l'utilisation [ENTITY-MIB](#) est quand vous avez seulement une CPU simple.

### [Exemple](#)

Surveillez l'utilisation de plusieurs CPU dans les 7500 châssis (RSP et deux VIPs). Le même applique aux linecards GSR. Utilisez la version du logiciel Cisco IOS 12.0(22)S3 ou plus tard quand vous votez c7500 ou GSR pour ces valeurs. Rendez-vous compte des bogues relatives : [CSCdw52978](#) (clients [enregistrés](#) seulement), [CSCdp17238](#) (clients [enregistrés](#) seulement).

1. Votez [cpmCPUTotal5min \(.1.3.6.1.4.1.9.9.109.1.1.1.5\)](#) pour obtenir « le pourcentage occupé global CPU pendant 5 derniers la période minute » pour toutes les CPU dans le châssis. La sortie prouve que le périphérique 7507 a trois CPU, utilisées pour 10%, 1%, et 2% pendant des 5 dernières minutes.

Router# **show processes CPU**

**CPU utilization for five seconds: 2%A/1%B; one minute: 1%C; five minutes: 1%D**

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>--

**Note:** Sur la base de la version du logiciel Cisco IOS qui fonctionne sur le périphérique, utilisez les objets appropriés MIB.

2. Afin d'identifier l'entité physique à laquelle ces valeurs correspondent, votez le [cpmCPUTotalPhysicalIndex \(.1.3.6.1.4.1.9.9.109.1.1.1.2\)](#). Vous voyez trois entités physiques avec les index 9, 25 et 28 :

Router# **show processes CPU**

**CPU utilization for five seconds: 2%A/1%B; one minute: 1%C; five minutes: 1%D**

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>--

3. Afin d'identifier la carte particulière à laquelle chaque entrée physique est connexe, entrée correspondante d'[entPhysicalName de balayage .1.3.6.1.2.1.47.1.1.1.7](#) (, avec les index précis 9, 25, 28 de l'étape 2, comme dernier chiffre. Vous voyez que RSP est utilisé pour 10%, et des VIPs dans les emplacements 4 et 6 sont utilisés pour un et deux pour cent.

Router# **show processes CPU**

CPU utilization for five seconds: 2%A/1%B; one minute: 1%C; five minutes: 1%D

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>--

## Procédure pour la CPU des cartes de RUELLE de Catalyst 5000

Le module LANE de Catalyst 5000 exécute le code IOS et prend en charge ces objets d'[OLD-CISCO-CPU-MIB](#) mentionné dans la [procédure pour des périphériques avec la CPU simple](#) :

- [busyPer](#)
- [avgBusy1](#)
- [avgBusy5](#)

Le module LANE de Catalyst 5000 n'a pas sa propre adresse IP. Par conséquent, vous devez utiliser l'adresse IP du superviseur de Catalyst avec l'indexation de chaîne de la communauté. Par exemple, si votre carte de RUELLE est dans l'emplacement 6 de votre Catalyst et votre chaîne de la communauté est publique, utilisez la chaîne "public@6" de la communauté et envoyez la demande SNMP à l'adresse IP du module de superviseur. Référez-vous au [pour en savoir plus d'indexation de chaîne de caractères de la communauté SNMP](#).

**Note:** Si vous ne spécifiez pas le numéro de module dans la communauté, vous recevez des données correspondant au premier module LANE dans le châssis du Catalyst.

## Informations connexes

- [Présentation des raisons d'un taux d'utilisation du processeur VIP à 99 % et présentation de la mise en mémoire tampon Rx-Side](#)
- [Dépannage de l'utilisation élevée du CPU sur les routeurs Cisco](#)
- [Indexation de la chaîne communautaire SNMP](#)
- [Localisateur MIB](#)
- [Index de protocole SNMP](#)
- [Support et documentation techniques - Cisco Systems](#)