

Surveiller l'utilisation du processeur sur la gamme ISR4300

Table des matières

[Introduction](#)

[Conditions préalables](#)

[Exigences](#)

[Composants utilisés](#)

[Architecture](#)

[Utilisation du processeur sur Cisco IOSd](#)

[Utilisation du processeur par trafic](#)

[Coeurs de processeur installés](#)

[Répartition des coeurs du processeur](#)

[Meilleures pratiques de surveillance du processeur](#)

Introduction

Ce document décrit comment lire l'utilisation de l'unité centrale (UC) sur les routeurs à services intégrés (ISR) de la gamme 4300.

Conditions préalables

Exigences

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Cisco IOS® XE
- ISR43XX

Composants utilisés

Les informations contenues dans ce document sont basées sur la version matérielle et logicielle :

- ISR4321/K9
- ISR4331/K9
- ISR4351/K9
- 03.16.01a.S // 15.5(3)S1a
- 03.16.04b.S // 15.5(3)S4b
- 16.9.7
- 16.12.4

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. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Architecture

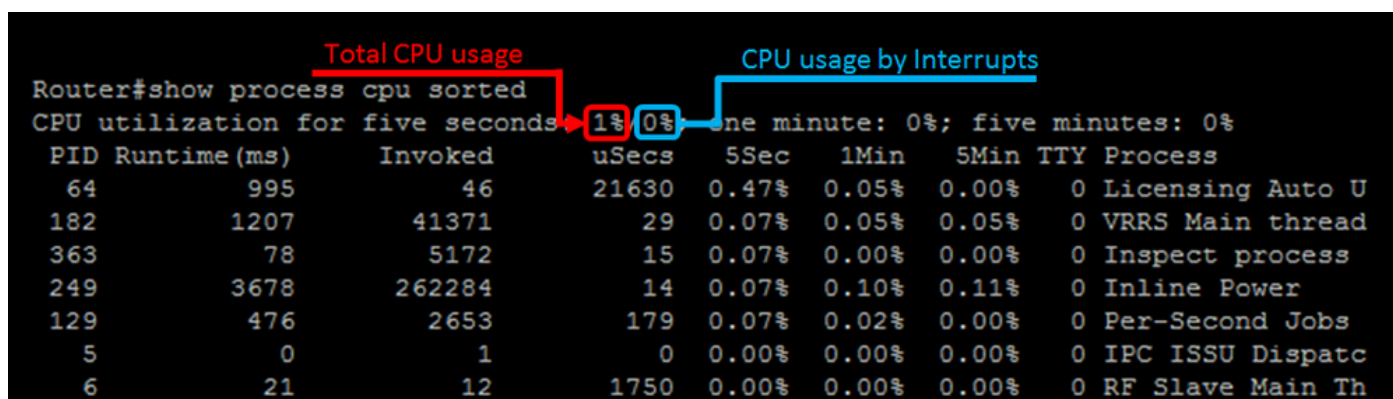
Les plates-formes de la gamme Cisco ISR 4000 exécutent Cisco IOS XE qui possède une architecture logicielle distribuée qui exécute un noyau Linux où Cisco IOS® s'exécute comme l'un des nombreux processus Linux. Cisco IOS s'exécute en tant que démon, appelé démon Cisco IOS (IOSd).

Utilisation du processeur sur Cisco IOSd

Afin de surveiller l'utilisation du CPU sur IOSd, exécutez la commande show process cpu :

```
#show process cpu
CPU utilization for five seconds: 1%/0%; one minute: 1%; five minutes: 0%
PID Runtime(ms)   Invoked    uSecs   5Sec   1Min   5Min TTY Process
  1         2         8        250  0.00%  0.00%  0.00%  0 Chunk Manager
  2         5        18        277  0.07%  0.00%  0.00%  0 Load Meter
  3         0         2         0   0.00%  0.00%  0.00%  0 DiagCard4/-1
  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
```

Le résultat affiche deux valeurs pour l'utilisation du CPU, la première valeur est la quantité totale d'utilisation du CPU et la seconde valeur est la quantité de CPU par les interruptions envoyées à IOSd :



```
Router#show process cpu sorted
CPU utilization for five seconds: 1% 0% one minute: 0%; five minutes: 0%
PID Runtime (ms)   Invoked    uSecs   5Sec   1Min   5Min TTY Process
  64         995         46    21630  0.47%  0.05%  0.00%  0 Licensing Auto U
 182        1207       41371     29  0.07%  0.05%  0.05%  0 VRRS Main thread
 363         78        5172     15  0.07%  0.00%  0.00%  0 Inspect process
 249        3678     262284     14  0.07%  0.10%  0.11%  0 Inline Power
 129         476       2653    179  0.07%  0.02%  0.00%  0 Per-Second Jobs
   5          0         1         0   0.00%  0.00%  0.00%  0 IPC ISSU Dispatc
   6          21        12    1750  0.00%  0.00%  0.00%  0 RF Slave Main Th
```

La différence entre la quantité totale de CPU et la quantité de CPU par interruptions sont les valeurs de CPU consommées par les processus ; afin de corroborer ajouter toute l'utilisation des processus pour les cinq dernières secondes :


- Consommation CPU des processus = 1 % - 0 % = 1 % = Tous les processus Consommation CPU indiquée dans la commande

Afin d'afficher les processus qui consomment le plus de CPU sur le dessus, exécutez la commande show process cpu sorted :

```
#show process cpu sorted
```


```
CPU utilization for five seconds: 1%/0%; one minute: 0%; five minutes: 0%
```

PID	Runtime(ms)	Invoked	uSecs	5Sec	1Min	5Min	TTY	Process
64	103	10	10300	0.33%	0.02%	0.00%	0	Licensing Auto U
83	26	231	112	0.27%	0.00%	0.00%	0	PuntInject Keepa
235	555	48176	11	0.11%	0.09%	0.07%	0	Inline Power
1	2	8	250	0.00%	0.00%	0.00%	0	Chunk Manager

 Remarque : l'ajout de tous les processus peut aboutir à des valeurs en virgule flottante, IOSd arrondit le résultat à l'entier suivant.

Utilisation du processeur par trafic

La conception de la gamme ISR4300, afin de transférer le trafic, est basée sur un élément appelé QuantumFlow Processor (QFP).

 Attention : QFP se trouve sur ASR1K comme une ou plusieurs puces physiques, sur l'ISR4400 la même fonctionnalité est faite avec les coprocesseurs Cavium Octeon, sur l'ISR4300 cette fonctionnalité est faite sur certains coeurs du processeur Intel principal. Vous pouvez considérer le QFP de la gamme ISR4300 comme un logiciel qui transfère les paquets.

Afin de déterminer la quantité de CPU consommée par le trafic, vous pouvez exécuter la commande `show platform hardware qfp active datapath use` :

```
#show platform hardware qfp active datapath utilization
CPP 0: Subdev 0
Input: Priority (pps)      5 secs      1 min      5 min      60 min
      (bps)              0           0           0           0
      Non-Priority (pps)  3           2           2           1
      (bps)             1448       992         992         568
      Total (pps)        3           2           2           1
      (bps)             1448       992         992         568
Output: Priority (pps)    0           0           0           0
      (bps)              0           0           0           0
      Non-Priority (pps)  3           2           2           1
      (bps)             12216      8024        8024        4576
      Total (pps)        3           2           2           1
      (bps)             12216      8024        8024        4576
Processing: Load (pct)   0           0           0           1
```

La commande répertorie l'utilisation du CPU en entrée et en sortie pour les paquets prioritaires et non prioritaires. Les informations sont affichées avec des paquets par seconde (PPS) et des bits par seconde (BPS). La dernière ligne affiche la quantité totale de charge du CPU due à l'acheminement des paquets en valeurs de pourcentage (PCT).

Coeurs de processeur installés

La gamme ISR4300 a une quantité différente de coeurs de CPU installés qui dépend du modèle, pour identifier le nombre de coeurs installés sur votre périphérique exécutez la commande show processes cpu platform :

```
#show processes cpu platform
CPU utilization for five seconds: 30%, one minute: 29%, five minutes: 29%
Core 0: CPU utilization for five seconds: 13%, one minute: 13%, five minutes: 13%
Core 1: CPU utilization for five seconds: 2%, one minute: 3%, five minutes: 3%
Core 2: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%
Core 3: CPU utilization for five seconds: 99%, one minute: 99%, five minutes: 99%
  Pid   PPid   5Sec   1Min   5Min  Status      Size  Name
-----
    1     0    0%    0%    0%  S          1863680  init
    2     0    0%    0%    0%  S              0  kthreadd
```

Vous pouvez également exécuter la commande show platform software status control-processor :

```
#show platform software status control-processor
<output omitted>
Per-core Statistics
CPU0: CPU Utilization (percentage of time spent)
  User: 4.80, System: 10.30, Nice: 0.00, Idle: 84.50
  IRQ: 0.40, SIRQ: 0.00, IOWait: 0.00
CPU1: CPU Utilization (percentage of time spent)
  User: 2.00, System: 3.40, Nice: 0.00, Idle: 94.59
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00
CPU2: CPU Utilization (percentage of time spent)
  User: 0.50, System: 0.00, Nice: 0.00, Idle: 99.49
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00
CPU3: CPU Utilization (percentage of time spent)
  User: 24.72, System: 75.27, Nice: 0.00, Idle: 0.00
  IRQ: 0.00, SIRQ: 0.00, IOWait: 0.00
```

D'autre part, exécutez la commande show platform software status control-processor brief, et l'une de ces commandes affiche la quantité de coeurs installés :

```
#show platform software status control-processor brief
<output omitted>
CPU Utilization
Slot CPU  User System  Nice  Idle  IRQ  SIRQ IOWait
RPO  0  4.30  9.80  0.00  85.90  0.00  0.00  0.00
     1  0.79  0.99  0.00  98.20  0.00  0.00  0.00
     2  0.50  0.00  0.00  99.50  0.00  0.00  0.00
     3 24.60 75.40  0.00  0.00  0.00  0.00  0.00
```

Répartition des coeurs du processeur

La conception de la gamme ISR4300 génère des coeurs spécifiques utilisés pour le traitement des paquets. Les coeurs 4 à 7 sont réservés au traitement des paquets sur ISR4331 et 4351, tandis que les coeurs 2 et 3 sont utilisés pour ISR4321.

Jusqu'à la version 16.5.x de Cisco IOS XE incluse pour des raisons de performances, le cadre de file d'attente hiérarchique (HQF) dispose d'un thread qui tourne toujours à chaud et s'exécute avec une utilisation CPU élevée, quelle que soit la configuration prête à l'emploi ou la quantité de trafic transitant par le système. Sur les plates-formes ISR4300, cela peut apparaître comme une utilisation CPU élevée sur un ou plusieurs coeurs, car le logiciel QFP s'exécute sur le processeur principal.

Cependant, après et avec les versions 16.6.x de Cisco IOS XE, une modification a été implémentée afin que ces plates-formes ne fassent pas tourner les threads à chaud. Dans ce cas, l'utilisation du CPU est plus distribuée par les coeurs.

Pour afficher l'utilisation du spin à chaud, exécutez la commande `show processes cpu platform sorted`, avant Cisco IOS XE 16.6.x :

```
#show processes cpu platform sorted
CPU utilization for five seconds: 28%, one minute: 29%, five minutes: 29%
Core 0: CPU utilization for five seconds: 12%, one minute: 13%, five minutes: 14%
Core 1: CPU utilization for five seconds: 2%, one minute: 3%, five minutes: 3%
Core 2: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%
Core 3: CPU utilization for five seconds: 99%, one minute: 99%, five minutes: 99% <<< hot-spin
  Pid   PPid   5Sec   1Min   5Min  Status      Size  Name
-----
  2541   1955   99%    99%    99%  S          1073807360  qfp-ucode-utah <<< high CPU process
  1551    929    7%     7%     7%  S          2038525952  fman_fp_image
```

Sur une architecture à huit coeurs, vous pouvez voir le même résultat, avec un coeur différent sur la rotation à chaud, avant Cisco IOS XE 16.6.x :


```
#show processes cpu platform sorted
CPU utilization for five seconds: 15%, one minute: 14%, five minutes: 15%
Core 0: CPU utilization for five seconds: 6%, one minute: 4%, five minutes: 8%
Core 1: CPU utilization for five seconds: 1%, one minute: 0%, five minutes: 2%
Core 2: CPU utilization for five seconds: 9%, one minute: 10%, five minutes: 7%
Core 3: CPU utilization for five seconds: 1%, one minute: 2%, five minutes: 1%
Core 4: CPU utilization for five seconds: 1%, one minute: 1%, five minutes: 1%
Core 5: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%
Core 6: CPU utilization for five seconds: 99%, one minute: 99%, five minutes: 99% <<< hot-spin
Core 7: CPU utilization for five seconds: 0%, one minute: 0%, five minutes: 0%
  Pid   PPid   5Sec   1Min   5Min  Status      Size  Name
-----
  3432   2779   99%    99%    99%  S          1086341120  qfp-ucode-utah <<< high CPU process
  2612   1893    7%     7%     7%  S          2038697984  fman_fp_image
  26114  25132    4%     5%     5%  R          42803200    hman
```

Après et y compris Cisco IOS XE 16.6.x, cependant vous pouvez voir qu'il y a une distribution de charge entre Core 2 et Core 3 :

```
----- show process cpu platform sorted -----  
CPU utilization for five seconds: 31%, one minute: 32%, five minutes: 29%  
Core 0: CPU utilization for five seconds: 3%, one minute: 3%, five minutes: 3%  
Core 1: CPU utilization for five seconds: 3%, one minute: 2%, five minutes: 2%  
Core 2: CPU utilization for five seconds: 39%, one minute: 41%, five minutes: 34% <<< load distributed  
Core 3: CPU utilization for five seconds: 84%, one minute: 83%, five minutes: 79% <<< load distributed  
  Pid   PPid   5Sec   1Min   5Min  Status      Size  Name  
-----  
26939  26344  127%   126%   116%  S           1195311104  qfp-ucode-utah <<< high CPU process
```

Après et y compris Cisco IOS XE 16.6.x, le même résultat que le précédent s'applique, mais pour les coeurs 4 à 7 :

```
----- show process cpu platform sorted -----  
CPU utilization for five seconds: 30%, one minute: 24%, five minutes: 27%  
Core 0: CPU utilization for five seconds: 41%, one minute: 13%, five minutes: 13%  
Core 1: CPU utilization for five seconds: 23%, one minute: 11%, five minutes: 13%  
Core 2: CPU utilization for five seconds: 19%, one minute: 10%, five minutes: 12%  
Core 3: CPU utilization for five seconds: 38%, one minute: 12%, five minutes: 12%  
Core 4: CPU utilization for five seconds: 28%, one minute: 26%, five minutes: 28% <<< load distributed  
Core 5: CPU utilization for five seconds: 53%, one minute: 40%, five minutes: 37% <<< load distributed  
Core 6: CPU utilization for five seconds: 18%, one minute: 16%, five minutes: 17% <<< load distributed  
Core 7: CPU utilization for five seconds: 93%, one minute: 81%, five minutes: 81% <<< load distributed  
  Pid   PPid   5Sec   1Min   5Min  Status      Size  Name  
-----  
26049  25462  164%   165%   170%  S           394128  qfp-ucode-utah <<< high CPU process
```

 Attention : si vous suspectez un problème d'utilisation du processeur principal, ouvrez un [dossier TAC \(Technical Assistance Center\)](#) afin d'obtenir de l'aide et de confirmer la stabilité du périphérique.

Meilleures pratiques de surveillance du processeur

Il est préférable d'utiliser les commandes spécifiques pour l'utilisation du chemin de données ou de l'IOSd. Le résultat des commandes d'affichage principales peut entraîner des alertes de faux positifs.

La commande permettant de surveiller l'utilisation du chemin de données est la suivante :

- show platform hardware qfp active datapath utilisation

La commande permettant de surveiller l'utilisation de l'IOSd est :

- show process cpu sorted

Utilisez l'un de ces Identificateurs d'objet (OID) pour surveiller l'utilisation du CPU IOSd avec le protocole SNMP (Simple Network Management Protocol) :

- [busyPer](#) = pourcentage d'occupation du processeur IOSd au cours des 5 dernières secondes
- [avgBusy1](#) = IOSd moyenne mobile exponentiellement décroissante d'une minute du pourcentage d'occupation du processeur
- [avgBusy5](#) = IOSd Moyenne mobile exponentiellement décroissante de cinq minutes du pourcentage d'occupation du processeur

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.