Utilisation élevée du CPU sur les commutateurs Catalyst 4500 basés sur le logiciel Cisco IOS

Contenu

Introduction

Les commutateurs de la gamme Catalyst 4500, qui incluent les commutateurs Catalyst 4948, sont dotés d'une méthodologie de gestion des paquets sophistiqués pour le trafic lié au CPU. Une utilisation CPU élevée sur ces commutateurs est un problème récurrent. Ce document fournit des détails sur l'architecture de gestion des paquets CPU et vous montre comment identifier les causes d'une utilisation CPU élevée sur ces commutateurs. Ce document mentionne également des scénarios courants de configuration ou de réseau qui entraînent une utilisation CPU élevée sur la gamme Catalyst 4500.

Remarque: Si vous utilisez des commutateurs de la gamme Catalyst 4500/4000 basés sur Catalyst OS (CatOS), référez vous au document Utilisation du CPU sur les commutateurs Catalyst 4500/4000, 2948G, 2980G, et 4912G qui exécutent le logiciel CatOS.

Conditions préalables

Conditions requises

Aucune spécification déterminée n'est requise pour ce document.

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Commutateurs de la gamme Catalyst 4500
- Commutateurs de la gamme Catalyst 4948

Remarque: Ce document applique seulement aux commutateurs fonctionnant sous Cisco IOS® et pas aux commutateurs sous CatOS.

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.

Informations générales

Avant que vous regardiez l'architecture de gestion de paquets CPU et dépanniez l'utilisation du CPU élevé, vous devez comprendre les différentes manières dans lesquelles les Commutateurs réalisés par matériel d'expédition et les Routeurs articulés autour d'un logiciel de Cisco IOS utilisent la CPU. On pense souvent, à tort, que l'utilisation CPU élevée indique l'épuisement des ressources sur un périphérique et la menace d'un crash. Un problème de capacité est l'un des symptômes de l'utilisation élevée du CPU sur des routeurs Cisco IOS. Cependant, un problème de capacité n'est presque jamais un symptôme d'une utilisation CPU élevée sur des commutateurs de transmissions matériels comme les commutateurs Catalyst 4500. Le commutateur Catalyst 4500 est conçu pour transférer des paquets dans l'ASIC matériel et atteindre des vitesses de transfert pouvant atteindre 102 millions de paquets par seconde (Mpps).

Le CPU du Catalyst 4500 remplit les fonctions suivantes :

- Gère les protocoles logiciels configurés, par exemple :Protocole Spanning Tree (STP)Protocole de routageCisco Discovery Protocol (CDP)Protocole d'agrégation de ports (PAgP)Protocole de jonction VLAN (VTP)Dynamic Trunking Protocol (DTP)
- Programme les entrées de configuration/dynamiques sur l'ASIC matériel, par exemple :Listes de contrôle d'accès (ACL)Entrées CEF
- Gère plusieurs composants en interne, par exemple :Cartes de ligne PoE (Power over Ethernet)Alimentations électriquesPlateau thermoventilateur
- Gère l'accès au commutateur, par exemple :TelnetConsoleProtocole de gestion de réseau simple (SNMP)
- Transfert les paquets par l'intermédiaire du chemin logiciel, par exemple :Paquets routés par

Internet Packet Exchange (IPX), uniquement pris en charge dans le chemin logicielFragmentation MTU (Maximum Transmission Unit)

Selon cette liste, l'utilisation CPU élevée peut résulter de la réception ou du traitement de paquets par le CPU. Certains des paquets qui sont envoyés pour traitement peuvent être essentiels pour le fonctionnement du réseau. Les unités BPDU (bridge protocol data unit) pour les configurations de topologie spanning tree. sont un exemple de ces paquets essentiels. Cependant, d'autre paquets peuvent être du trafic de données transmis par logiciel. Ces scénarios exigent que l'ASIC de commutation envoie des paquets au CPU pour traitement :

- Paquets copiés dans le CPU, mais dont les paquets d'origine sont commutés dans le matérielUn exemple est l'apprentissage des adresses hôtes MAC.
- Paquets envoyés au CPU pour traitementExemples :Mises à jour du protocole de routageBPDUUn flux de trafic volontaire ou involontaire
- Paquets envoyés au CPU pour le transfertPar exemple, les paquets qui nécessitent le routage IPX ou AppleTalk.

Comprendre l'architecture de la gestion des paquets du CPU du commutateur Catalyst 4500

Le Catalyst 4500 dispose d'un mécanisme de qualité de service intégré (QoS) afin de différencier les types trafic destinés au CPU. Ce mécanisme différencie le trafic en fonction des informations de paquet de la couche 2 (L2)/couche 3 (L3)/couche 4(L4). Le moteur de superviseur de paquets a 16 files d'attente afin de gérer plusieurs types de paquets ou événements. La figure 1 présente ces files d'attente. Le tableau 1 répertorie les files d'attente et les types de paquet qu'elles contiennent. Les 16 files d'attente permettent au Catalyst 4500 de mettre les paquets en attente en fonction du type du paquet et de sa priorité.

Figure 1 - Le commutateur Catalyst 4500 utilise plusieurs files d'attente CPU

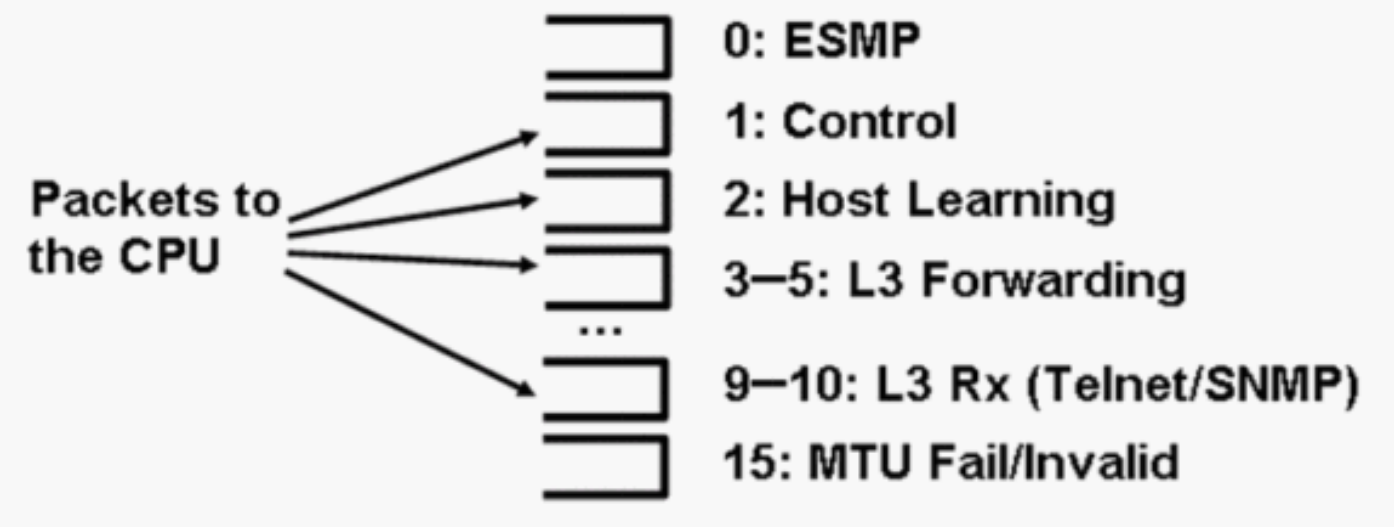


Tableau 1 - Description de la file d'attente du Catalyst 4500

Numéro de la file	Nom de la file d'attente	Paquets mis dans la file d'attente

d'attente		
0	Esmp	[1]paquet ESMP (paquets de gestion internes) pour le linecard ASIC ou toute autre Gestion composante
1	Contrôle	Paquets d'avion du contrôle L2, tels que STP, CDP, PAgP, LACP[2], ou UDLD[3]
2	Apprentissage d'hôte	Trames avec adresses source MAC inconnues qui sont copiées vers le CPU afin de construire la table de transfert L2
3, 4, 5	L3 Fwd Highest, L3 Fwd High/Medium, L3 Fwd Low	Les paquets qui doivent être expédiés en logiciel, tel que GRE[4] perce un tunnel si l'ARP[5] est non résolu pour l'adresse IP de destination, des paquets sont envoyés à cette file d'attente.
6, 7, 8	L2 Fwd Highest, L2 Fwd High/Medium, L2 Fwd Low	Paquets transférés à la suite d'un pontage • Les protocoles non pris en charge dans le matériel, tels que les paquets routés IPX et AppleTalk sont pontés vers le CPU • Requête et réponse ARP • Des paquets avec une adresse MAC de destination de l'interface du commutateur SVI[6/L3] pont si les paquets ne peuvent pas être conduits dans le matériel en raison de : Options d'en-tête IPTTL expiré[7]Encapsulation non-ARPA
9, 10	L3 Rx High, L3 Rx Low	Le trafic d'avion du contrôle L3, par exemple, des protocoles de routage, qui est destiné aux exemples d'adresses IP CPU incluent le telnet, le SNMP, et le SSH[8].
11	Échec RPF	Paquets de multidiffusion qui ont manqué le contrôle RPF[9]
12	ACL fwd(snooping)	Paquets qui sont traités par le DHCP[10] pillant, inspection dynamique d'ARP, ou caractéristiques pillantes IGMP[11]
13	ACL log, unreach	Les paquets qui ont frappé ACE[12] avec le **mot clé de journal** ou les paquets qui étaient dus relâché à un refuser dans un ACL de sortie ou au manque d'une artère à la destination ces paquets exigent la génération des messages ICMP inaccessibles.

14	ACL sw processing	Paquets qui sont donnés un coup de volée à la CPU due à un manque de ressources en matériel supplémentaires d'ACL, telles que TCAM[13], pour l'ACL de Sécurité
15	MTU Fail/Invalid	Paquets devant être fragmentés car l'interface de sortie MTU est plus petite que le paquet.

[1] ESMP = même protocole de gestion simple.

[2] LACP = Control Protocol d'agrégation de liaisons.

[3] UDLD = détection unidirectionnelle de lien.

[4] GRE = encapsulation générique de routage.

ARP [5] = Address Resolution Protocol.

[6] SVI = interface virtuelle commutée.

[7] TTL = Time to Live.

SSH [8] = Secure Shell Protocol.

[9] RPF = Reverse Path Forwarding

[10] DHCP = protocole DHCP.

[11] IGMP = protocole de gestion de groupes Internet (IGMP).

[12] ACE = entrée de contrôle d'accès.

[13] TCAM = mémoire associative ternaire.

Les files d'attente ci-dessous sont des files d'attente distinctes :

- L2 Fwd Highest ou L3 Fwd Highest
- L2 Fwd High/Medium ou L3 Fwd High/Medium
- L2 Fwd Low ou L3 Fwd Low
- L3 Rx High ou L3 Rx Low

Les paquets sont placés dans ces files d'attente en fonction de l'étiquette QoS, qui est la valeur DSCP (Differentiated Services Code Poin) du type de service IP (ToS). Par exemple, les paquets avec un DSCP de 63 sont placés dans la file d'attente L3 Fwd Highest. Vous pouvez voir les paquets reçus et perdus pour ces 16 files d'attente dans la sortie de la commande **show platform cpu packet statistics all**. La sortie de cette commande est très longue. Émettez la commande **show platform cpu packet statistics** afin d'afficher uniquement les événements non nuls. La commande **show platform cpuport** constitue une commande alternative. Utilisez uniquement la commande **show platform cpuport** si vous exécutez le logiciel Cisco IOS Version 12.1(11)EW ou antérieure. Cette commande est maintenant obsolète. Cependant, cette commande plus ancienne faisait partie de la commande **show tech-support** dans des versions du logiciel Cisco IOS antérieures au logiciel Cisco IOS Version 12.2(20)EWA.

Utilisez la commande **show platform cpu packet statistics** pour tout type de dépannage.

```
Switch#show platform cpu packet statistics all
!--- Output suppressed. Total packet queues 16 Packets Received by Packet Queue Queue Total 5
sec avg 1 min avg 5 min avg 1 hour avg ---------------------- -------------- --------- --------
- --------- ---------- Esmp 0 0 0 0 0 Control 48 0 0 0 0 Host Learning 0 0 0 0 0 L3 Fwd High 0 0
0 0 0 L3 Fwd Medium 0 0 0 0 0 L3 Fwd Low 0 0 0 0 0 L2 Fwd High 0 0 0 0 0 L2 Fwd Medium 0 0 0 0 0
L2 Fwd Low 0 0 0 0 0 L3 Rx High 0 0 0 0 0 L3 Rx Low 0 0 0 0 0 RPF Failure 0 0 0 0 0 ACL
fwd(snooping) 0 0 0 0 0 ACL log, unreach 0 0 0 0 0 ACL sw processing 0 0 0 0 0 MTU Fail/Invalid
0 0 0 0 0 Packets Dropped by Packet Queue Queue Total 5 sec avg 1 min avg 5 min avg 1 hour avg -
---------------------- -------------- --------- --------- --------- ---------- Esmp 0 0 0 0 0
Control 0 0 0 0 0 Host Learning 0 0 0 0 0 L3 Fwd High 0 0 0 0 0 L3 Fwd Medium 0 0 0 0 0 L3 Fwd
Low 0 0 0 0 0 L2 Fwd High 0 0 0 0 0 L2 Fwd Medium 0 0 0 0 0 L2 Fwd Low 0 0 0 0 0 L3 Rx High 0 0
0 0 0 L3 Rx Low 0 0 0 0 0 RPF Failure 0 0 0 0 0 ACL fwd(snooping) 0 0 0 0 0 ACL log, unreach 0 0
0 0 0 ACL sw processing 0 0 0 0 0 MTU Fail/Invalid 0 0 0 0 0
```

Le CPU du Catalyst 4500 pondère les diverses files d'attente que le tableau 1 répertorie. Le CPU attribue une pondération en fonction de l'importance, du type et de la priorité du trafic ou de DSCP. Le CPU traite la file d'attente en fonction du poids relatif de la file d'attente. Par exemple, si un paquet de contrôle, tel qu'un BPDU, et une demande d'écho ICMP sont en attente, le CPU traite d'abord le paquet de contrôle. Une quantité excessive de trafic non prioritaire ou moins important n'empêche pas le CPU de pouvoir traiter ou gérer le système. Ce mécanisme garantit que le réseau reste stable même lors d'une utilisation CPU élevée. Cette capacité du réseau à rester stable constitue une information essentielle que vous devez comprendre.

Il existe un autre détail très important concernant la mise en œuvre de la gestion des paquets par le CPU du Catalyst 4500. Si le CPU a déjà traité des paquets ultra-prioritaires ou des processus mais ne dispose plus de cycles CPU disponibles pendant une période en particulier, le CPU gère les paquets non prioritaires de la file d'attente ou exécute en arrière-plan des processus d'une priorité plus basse. L'utilisation CPU élevée causée par le traitement de paquets non prioritaires ou de processus en arrière-plan est normale car le CPU tente constamment d'utiliser toutes les ressources disponibles. De cette façon, le CPU tente d'obtenir des performances maximales pour le commutateur et le réseau sans sacrifier la stabilité du commutateur. Le commutateur Catalyst 4500 considère que le CPU est sous-utilisé à moins que le CPU soit utilisé à 100 pourcent pour un seul intervalle de temps.

Le logiciel Cisco IOS Version 12.2(25)EWA2 et ultérieure ont amélioré le mécanisme de gestion des paquets CPU et des processus et de comptage. Par conséquent, utilisez ces versions sur vos déploiements Catalyst 4500.

Identifier la raison de l'utilisation CPU élevée sur le commutateur Catalyst 4500

Maintenant que vous comprenez l'architecture et la conception de gestion de paquets du CPU, vous souhaitez peut-être déterminer pourquoi l'utilisation du CPU de votre Catalyst 4500 est élevée. Le Catalyst 4500 dispose des commandes et des outils nécessaires pour identifier la cause principale de l'utilisation CPU élevée. Une fois cette raison identifiée, les administrateurs peuvent exécuter l'une ou l'autre de ces actions :

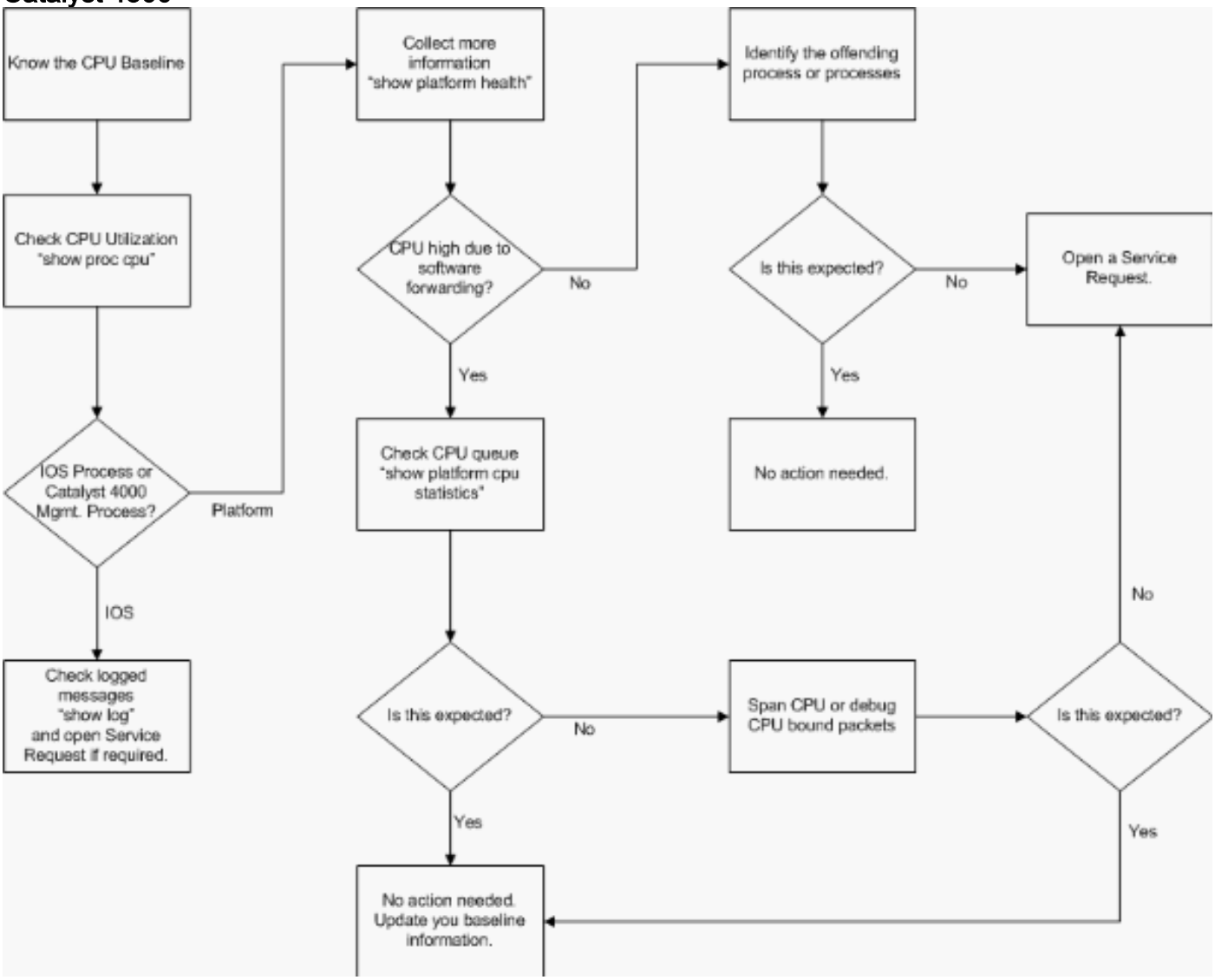
- Action corrective - Cela peut comprendre des modifications de la configuration ou du réseau ou la création d'une demande de service auprès de l'assistance technique Cisco pour une analyse approfondie.
- Aucune action - Le Catalyst 4500 fonctionne conformément aux attentes. Le CPU affiche une utilisation élevée car le moteur de superviseur optimise les cycles du CPU afin d'effectuer tous

les transferts logiciels de paquets et travaux d'arrière-plan nécessaires.

Assurez-vous d'avoir identifié la cause d'une utilisation élevée du CPU, même si une action corrective n'est pas nécessaire dans tous les cas. Une utilisation CPU élevée peut simplement être le symptôme d'un problème sur le réseau. La résolution de la cause principale de ce problème peut être nécessaire afin de réduire l'utilisation du CPU.

La figure 2 présente la méthodologie de dépannage à utiliser afin d'identifier la cause principale d'une utilisation CPU élevée sur le Catalyst 4500.

Figure 2 - Méthodologie de dépannage d'une utilisation CPU élevée sur les commutateurs Catalyst 4500



Les étapes de dépannage générales sont les suivantes :

1. Émettez la commande **show processes cpu** afin d'identifier les processus de Cisco IOS qui utilisent des cycles CPU.
2. Émettez la commande **show platform health** afin d'identifier les processus spécifiques à la plate-forme.
3. Si le processus très actif est **K2CpuMan Review**, émettez la commande **show platform cpu packet statistics** afin d'identifier le type de trafic qui atteint le CPU.Si l'activité n'est pas due à **K2CpuMan Review**, ignorez l'étape 4 et passez à l'étape 5.
4. Identifiez les paquets qui atteignent le CPU en utilisant les outils de dépannage d'analyse du trafic destiné au CPU, si nécessaire.Le Switched Port Analyzer (SPAN) est un exemple

d'outil de dépannage à utiliser.

5. Passez en revue ce document ainsi que la section Dépanner les problèmes courants liés à une utilisation CPU élevée pour en connaître les causes courantes.Si vous ne parvenez toujours pas à identifier la cause principale, contactez l'assistance technique Cisco.

Spécification de base pour l'utilisation CPU

La première étape importante est de connaître l'utilisation CPU de votre commutateur pour votre configuration et la configuration du réseau. Utilisez la commande **show processes cpu** afin d'identifier l'utilisation CPU sur le commutateur Catalyst 4500. Une mise à jour constante des spécifications de base pour l'utilisation du CPU peut être nécessaire lorsque vous rendez la configuration du réseau plus complexe ou lorsque votre modèle de trafic réseau change. La figure 2 indique cette nécessité.

Cette sortie provient d'un commutateur Catalyst 4507R complètement chargé. L'état d'équilibre du CPU se situe entre 32 et 38 pourcent, ce qui est nécessaire afin de remplir les fonctions de gestion pour ce commutateur :

```
Switch#show processes cpu
CPU utilization for five seconds: 38%/1%; one minute: 32%; five minutes: 32%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           0        63          0  0.00%  0.00%  0.00%   0 Chunk Manager
   2          60     50074          1  0.00%  0.00%  0.00%   0 Load Meter
   3           0         1          0  0.00%  0.00%  0.00%   0 Deferred Events
!--- Output suppressed. 27 524 250268 2 0.00% 0.00% 0.00% 0 TTY Background 28 816 254843 3 0.00%
0.00% 0.00% 0 Per-Second Jobs 29 101100 5053 20007 0.00% 0.01% 0.00% 0 Per-minute Jobs 30
26057260  26720902        975 12.07% 11.41% 11.36%   0 Cat4k Mgmt HiPri
  31    19482908  29413060        662 24.07% 19.32% 19.20%   0 Cat4k Mgmt LoPri
  32        4468    162748         27  0.00%  0.00%  0.00%   0 Galios Reschedul
  33           0         1          0  0.00%  0.00%  0.00%   0 IOS ACL Helper
  34           0         2          0  0.00%  0.00%  0.00%   0 NAM Manager
```

Une utilisation CPU de cinq secondes est exprimée comme suit :

```
Switch#show processes cpu
CPU utilization for five seconds: 38%/1%; one minute: 32%; five minutes: 32%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           0        63          0  0.00%  0.00%  0.00%   0 Chunk Manager
   2          60     50074          1  0.00%  0.00%  0.00%   0 Load Meter
   3           0         1          0  0.00%  0.00%  0.00%   0 Deferred Events
!--- Output suppressed. 27 524 250268 2 0.00% 0.00% 0.00% 0 TTY Background 28 816 254843 3 0.00%
0.00% 0.00% 0 Per-Second Jobs 29 101100 5053 20007 0.00% 0.01% 0.00% 0 Per-minute Jobs 30
26057260  26720902        975 12.07% 11.41% 11.36%   0 Cat4k Mgmt HiPri
  31    19482908  29413060        662 24.07% 19.32% 19.20%   0 Cat4k Mgmt LoPri
  32        4468    162748         27  0.00%  0.00%  0.00%   0 Galios Reschedul
  33           0         1          0  0.00%  0.00%  0.00%   0 IOS ACL Helper
  34           0         2          0  0.00%  0.00%  0.00%   0 NAM Manager
```

x% représente l'utilisation totale du CPU et *y%* représente le CPU utilisé au niveau d'interruption. Lorsque vous dépannez les commutateurs Catalyst 4500, concentrez-vous uniquement sur l'utilisation totale du CPU.

Comprendre la commande show processes cpu sur les commutateurs Catalyst 4500

Cette sortie de **show processes cpu** montre que deux processus utilisent le CPU : **Cat4k Mgmt HiPri** et **Cat4k Mgmt LoPri**. Ces deux processus regroupent plusieurs processus spécifiques à une

plate-forme qui remplissent les fonctions de gestion essentielles sur le Catalyst 4500. Ces processus traitent des plans de contrôle aussi bien que des paquets de données devant être commutés ou traités de manière logicielle.

Afin de voir quels processus spécifiques à une plate-forme utilisent le CPU dans le contexte de **Cat4k Mgmt HiPri** et de **Cat4k Mgmt LoPri**, émettez la commande **show platform health**.

Chacun des processus spécifiques à une plate-forme a une utilisation cible/prévue du CPU. Lorsque ce processus fait partie de la cible, le CPU l'exécute dans le contexte hautement prioritaire. La sortie de la commande **show processes cpu** compte cette utilisation sous **Cat4k Mgmt HiPri**. Si un processus dépasse l'utilisation cible/prévue, il s'exécute dans le contexte non prioritaire. La sortie de la commande **show processes cpu** compte cette utilisation supplémentaire sous **Cat4k Mgmt LoPri**. Ce **Cat4k Mgmt LoPri** est également utilisé pour exécuter des processus d'arrière-plan et d'autres processus non prioritaires, tels que le contrôle de cohérence et la lecture des compteurs d'interface. Ce mécanisme permet au CPU d'exécuter des processus hautement prioritaires si nécessaire, et les cycles CPU restants sont utilisés pour les processus non prioritaires. Un léger dépassement de l'utilisation cible du CPU ou un pic d'utilisation momentané n'indiquent pas un problème nécessitant une enquête.

```
Switch#show platform health
                     %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                     Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
Lj-poll                1.00   0.02      2      1  100  500    0   0    0  1:09
GalChassisVp-review    3.00   0.29     10      3  100  500    0   0    0  11:15
S2w-JobEventSchedule  10.00   0.32     10      7  100  500    0   0    0  10:14
Stub-JobEventSchedul  10.00  12.09     10      6  100  500   14  13    9  396:35
StatValueMan Update    1.00   0.22      1      0  100  500    0   0    0  6:28
Pim-review             0.10   0.00      1      0  100  500    0   0    0  0:22
Ebm-host-review        1.00   0.00      8      0  100  500    0   0    0  0:05
Ebm-port-review        0.10   0.00      1      0  100  500    0   0    0  0:01
Protocol-aging-revie   0.20   0.00      2      0  100  500    0   0    0  0:00
Acl-Flattener  e       1.00   0.00     10      0  100  500    0   0    0  0:00
KxAclPathMan create/   1.00   0.00     10      5  100  500    0   0    0  0:39
KxAclPathMan update    2.00   0.00     10      0  100  500    0   0    0  0:00
KxAclPathMan reprogr   1.00   0.00      2      0  100  500    0   0    0  0:00
TagMan-RecreateMtegR   1.00   0.00     10      0  100  500    0   0    0  0:00
K2CpuMan Review       30.00  10.19     30     28  100  500   14  13    9  397:11
K2AccelPacketMan: Tx  10.00   2.20     20      0  100  500    2   2    1  82:06
K2AccelPacketMan: Au   0.10   0.00      0      0  100  500    0   0    0  0:00
K2AclMan-taggedFlatA   1.00   0.00     10      0  100  500    0   0    0  0:00
K2AclCamMan stale en   1.00   0.00     10      0  100  500    0   0    0  0:00
K2AclCamMan hw stats   3.00   1.04     10      5  100  500    1   1    0  39:36
K2AclCamMan kx stats   1.00   0.00     10      5  100  500    0   0    0  13:40
K2AclCamMan Audit re   1.00   0.00     10      5  100  500    0   0    0  13:10
K2AclPolicerTableMan   1.00   0.00     10      1  100  500    0   0    0  0:38
K2L2 Address Table R   2.00   0.00     12      5  100  500    0   0    0  0:00
K2L2 New Static Addr   2.00   0.00     10      1  100  500    0   0    0  0:00
K2L2 New Multicast A   2.00   0.00     10      5  100  500    0   0    0  0:01
K2L2 Dynamic Address   2.00   0.00     10      0  100  500    0   0    0  0:00
K2L2 Vlan Table Revi   2.00   0.00     12      9  100  500    0   0    0  0:01
K2 L2 Destination Ca   2.00   0.00     10      0  100  500    0   0    0  0:00
K2PortMan Review       2.00   0.72     15     11  100  500    1   1    0  37:22
Gigaport65535 Review   0.40   0.07      4      2  100  500    0   0    0  3:38
Gigaport65535 Review   0.40   0.08      4      2  100  500    0   0    0  3:39
K2Fib cam usage revi   2.00   0.00     15      0  100  500    0   0    0  0:00
K2Fib IrmFib Review    2.00   0.00     15      0  100  500    0   0    0  0:00
K2Fib Vrf Default Ro   2.00   0.00     15      0  100  500    0   0    0  0:00
K2Fib AdjRepop Revie   2.00   0.00     15      0  100  500    0   0    0  0:00
K2Fib Vrf Unpunt Rev   2.00   0.01     15      0  100  500    0   0    0  0:23
K2Fib Consistency Ch   1.00   0.00      5      2  100  500    0   0    0  29:25
```

```
K2FibAdjMan Stats Re   2.00   0.30     10      4  100  500    0   0    0  6:21
K2FibAdjMan Host Mov   2.00   0.00     10      4  100  500    0   0    0  0:00
K2FibAdjMan Adj Chan   2.00   0.00     10      0  100  500    0   0    0  0:00
K2FibMulticast Signa   2.00   0.01     10      2  100  500    0   0    0  2:04
K2FibMulticast Entry   2.00   0.00     10      7  100  500    0   0    0  0:00
K2FibMulticast Irm M   2.00   0.00     10      7  100  500    0   0    0  0:00
K2FibFastDropMan Rev   2.00   0.00      7      0  100  500    0   0    0  0:00
K2FibPbr route map r   2.00   0.06     20      5  100  500    0   0    0  16:42
K2FibPbr flat acl pr   2.00   0.07     20      2  100  500    0   0    0  3:24
K2FibPbr consolidati   2.00   0.01     10      0  100  500    0   0    0  0:24
K2FibPerVlanPuntMan    2.00   0.00     15      4  100  500    0   0    0  0:00
K2FibFlowCache flow    2.00   0.01     10      0  100  500    0   0    0  0:23
K2FibFlowCache flow    2.00   0.00     10      0  100  500    0   0    0  0:00
K2FibFlowCache adj r   2.00   0.01     10      0  100  500    0   0    0  0:20
K2FibFlowCache flow    2.00   0.00     10      0  100  500    0   0    0  0:06
K2MetStatsMan Review   2.00   0.14      5      2  100  500    0   0    0  23:40
K2FibMulticast MET S   2.00   0.00     10      0  100  500    0   0    0  0:00
K2QosDblMan Rate DBL   2.00   0.12      7      0  100  500    0   0    0  4:52
IrmFibThrottler Thro   2.00   0.01      7      0  100  500    0   0    0  0:21
K2 VlanStatsMan Revi   2.00   1.46     15      7  100  500    2   2    1  64:44
K2 Packet Memory Dia   2.00   0.00     15      8  100  500    0   1    1  45:46
K2 L2 Aging Table Re   2.00   0.12     20      3  100  500    0   0    0  7:22
RkiosPortMan Port Re   2.00   0.73     12      7  100  500    1   1    1  52:36
Rkios Module State R   4.00   0.02     40      1  100  500    0   0    0  1:28
Rkios Online Diag Re   4.00   0.02     40      0  100  500    0   0    0  1:15
RkiosIpPbr IrmPort R   2.00   0.02     10      3  100  500    0   0    0  2:44
RkiosAclMan Review     3.00   0.06     30      0  100  500    0   0    0  2:35
MatMan Review          0.50   0.00      4      0  100  500    0   0    0  0:00
Slot 3 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 3 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 4 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 4 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 5 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 5 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 6 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 6 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 7 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 7 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
EthHoleLinecardMan(1   1.66   0.04     10      0  100  500    0   0    0  1:18
EthHoleLinecardMan(2   1.66   0.02     10      0  100  500    0   0    0  1:18
EthHoleLinecardMan(6   1.66   0.17     10      6  100  500    0   0    0  6:38
                     -------------
%CPU Totals          212.80  35.63
```

Comprendre la commande show platform health sur les commutateurs Catalyst 4500

La commande **show platform health** fournit beaucoup d'informations pertinentes uniquement pour un ingénieur de développement. Afin de dépanner une utilisation CPU élevée, recherchez un chiffre élevé dans la colonne %CPU actual de la sortie. Observez également les éléments affichés à droite de cette colonne afin de vérifier l'utilisation CPU de ce processus dans les colonnes 1 minute et 1 heure average %CPU. Parfois, les processus connaissent un pic momentané sans utiliser le CPU très longtemps. Une partie de l'utilisation élevée momentanée du CPU se produit lors de la programmation du matériel ou l'optimisation de la programmation. Par exemple, un pic de l'utilisation CPU est normal lors de la programmation matérielle d'une grande ACL dans le TCAM.

Dans la sortie de la commande **show platform health** de la section Comprendre la commande show platform health sur les commutateurs Catalyst 4500, les processus **Stub-JobEventSchedul** et **K2CpuMan Review** utilisent un nombre élevé de cycles CPU. Le Tableau 2 fournit certaines

informations de base concernant les processus spécifiques à une plate-forme courants qui apparaissent dans la sortie de la commande **show platform health**.

Tableau 2 - Description des processus spécifiques à une plate-forme présentés par la commande show platform health

Nom du processus spécifique à une plate-forme	Description
Pim-review	Gestion de l'état du châssis/de la carte de ligne
Ebm	Module de pont Ethernet, tel que le vieillissement et la surveillance
Acl-Flattener / **K2AclMan**	Processus de fusion ACL
KxAclPathMan - TagMan-examen de chemin	Gestion et maintenance d'état d'ACL
K2CpuMan Review	Le processus qui exécute le transfert de paquet de logiciel si vous voyez l'utilisation du CPU élevé due à ce processus, étudient les paquets qui frappent la CPU avec l'utilisation de la commande de **statistiques de paquet CPU de show platform**.
K2AccelPacketMan	Pilote qui interagit avec le moteur de paquet afin d'envoyer des paquets envoyés depuis le CPU
K2AclCamMan	Gère le matériel d'entrée et de sortie TCAM pour QoS et les fonctions de sécurité
K2AclPolicerTableMan	Contrôle les applicateurs de stratégies d'entrée et sortie
K2L2	Représente le sous-système de l'expédition L2 du logiciel de Cisco IOS du Catalyst 4500 que ces processus sont responsables de la maintenance des diverses tables L2.
K2PortMan Review	Gère les fonctions de programmation liées aux ports
K2Fib	Gestion du FIB[1]
K2FibFlowCache	Gestion de cache PBR[2]
K2FibAdjMan	Gestion de la table de juxtaposition FIB
K2FibMulticast	Gère les entrées FIB Multicast
K2MetStatsMan Review	Gère A RENCONTRÉ les statistiques[3]
K2QosDblMan Review	Gère le qos dbl[4]

IrmFibThrottler Thro	Module de routage IP
K2 L2 Aging Table Re	Gère la fonction de vieillissement L2
GalChassisVp-review	Surveillance de l'état du châssis
S2w-JobEventSchedule	Parvient les protocoles S2W[5] pour surveiller l'état de linecards
Stub-JobEventSchedul	Surveillance et maintenance des cartes de ligne de remplacement basées sur ASIC
RkiosPortMan Port Re	Surveillance et maintenance de l'état des ports
Rkios Module State R	Surveillance et maintenance des cartes de ligne
EthHoleLinecardMan	Gère GBIC[6] dans chacun des linecards

[1] FIB = Forwarding Information Base.

[2] PBR = routage basé sur la politique.

[3] RÉUNI = Tableau d'extension de Multidiffusion.

DBL [4] = limitation dynamique de mémoire tampon.

[5] S2W = séquentiel-à-fil.

[6] GBIC = convertisseur d'interface de gigabit.

Dépanner les problèmes courants liés à une utilisation CPU élevée

Cette section traite de certains des problèmes courants liés à une utilisation CPU élevée sur les commutateurs Catalyst 4500.

Utilisation CPU élevée due aux paquets commutés par processus

Une des raisons courantes d'une utilisation CPU élevée est que le CPU du Catalyst 4500 est occupé par le traitement des paquets transmis par logiciel ou des paquets de contrôle. Les paquets IPX ou les paquets de contrôle, tels que BPDU constituent des exemples de paquets transmis par logiciel. Une petite partie de ces paquets est généralement envoyée au CPU. Cependant, un nombre de paquets régulièrement important peut indiquer une erreur de configuration ou un événement sur le réseau. Vous devez identifier la cause des événements qui mènent au transfert de paquets au CPU pour traitement. Cette identification vous permet de déboguer les problèmes liés à une utilisation CPU élevée.

Raisons courantes d'une utilisation CPU élevée due aux paquets à commutation par processus :

- Un nombre élevé d'exemples de port de spanning tree

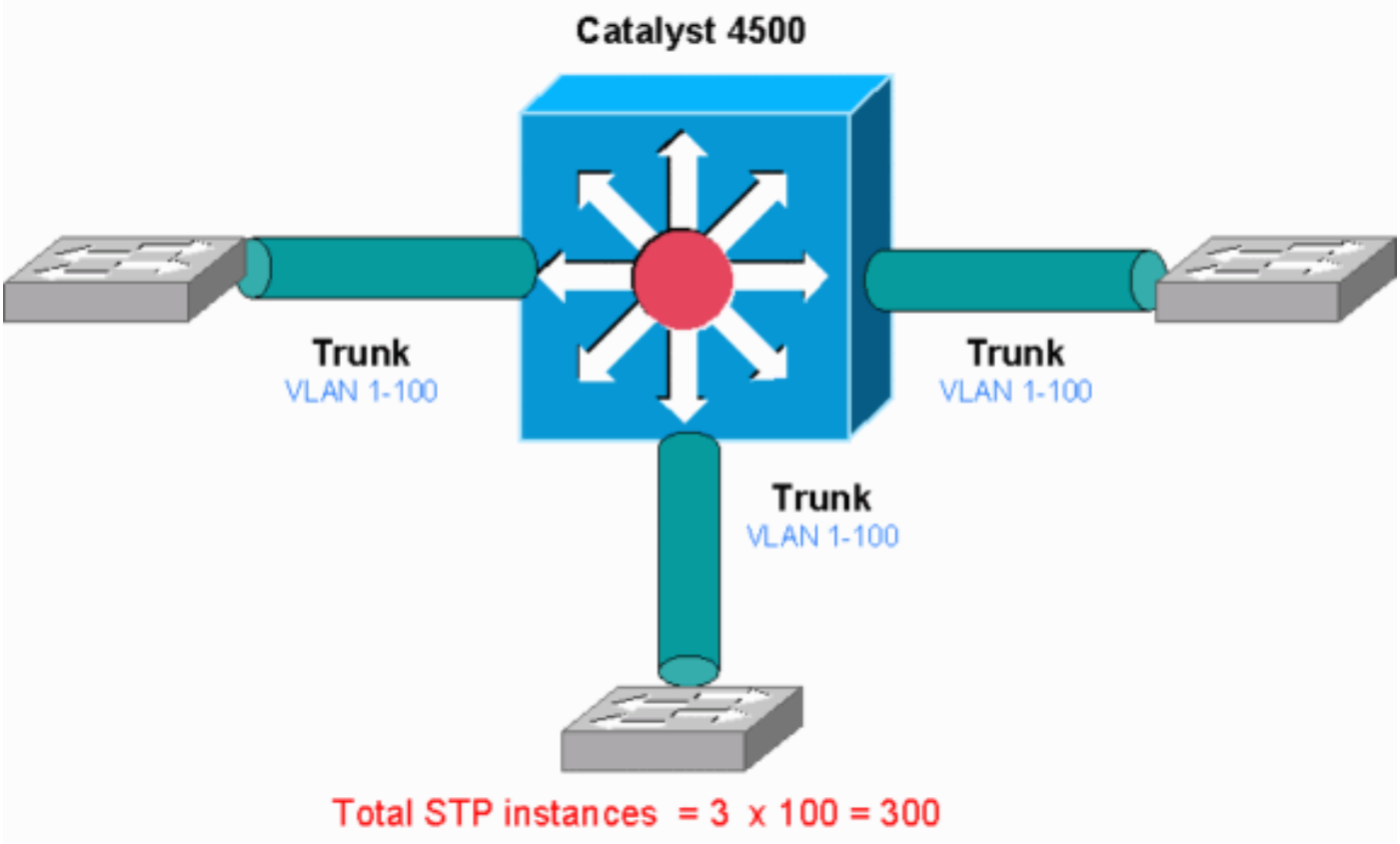
- Redirections ICMP ; routage de paquets sur la même interface
- Routage IPX ou AppleTalk
- Apprentissage d'hôte
- Manque de ressources matérielles (TCAM) pour la sécurité de liste de contrôle d'accès
- Le mot clé de journal dans l'ACL
- Boucles de transfert de la couche 2

Autres raisons de la commutation de paquets sur le CPU :

- Fragmentation MTU - Assurez-vous que toutes les interfaces le long du chemin du paquet ont la même MTU.
- ACL avec des indicateurs TCP autres qu'**établi**
- Routage IP version 6 (IPv6) - Pris en charge uniquement par l'intermédiaire du chemin de commutation logiciel.
- GRE — Ceci est pris en charge seulement par l'intermédiaire du chemin de logiciel-commutation.
- Refus du trafic dans l'ACL du routeur (RACL) entrant ou sortant**Remarque:** Le débit est limité dans le logiciel Cisco IOS Version 12.1(13)EW1 et ultérieure.Émettez la commande **no ip unreachables** sous interface de l'ACL.
- Un trafic ARP et DHCP excessif arrive au CPU pour traitement en raison d'un grand nombre de serveurs directement connectésSi vous suspectez une attaque DHCP, utilisez le DCHP snooping pour limiter le débit du trafic DHCP depuis n'importe quel port hôte spécifique.
- Nombre de requêtes SNMP excessif par une station d'extrémité légitime ou présentant un comportement inattendu

Un nombre élevé d'instances de port spanning-tree

Le Catalyst 4500 prend en charge 3 000 instances de ports ou ports actifs de spanning tree en mode Per VLAN spanning-tree + (PVST+). Tous les moteurs de superviseur sont pris en charge à l'exception de Supervisor Engine II+ et II+TS, et de Catalyst 4948. Supervisor Engine II+ et II+TS, et Catalyst 4948 prennent en charge jusqu'à 1 500 instances de port. Si vous dépassez ces recommandations d'instance STP, le commutateur présente une utilisation CPU élevée.



Ce diagramme présente un commutateur Catalyst 4500 avec trois ports de liaison qui transportent chacun les VLAN 1 à 100. Cela équivaut à 300 instances de port de spanning-tree. Généralement vous pouvez calculer des instances de port de spanning-tree avec cette formule :

```
Switch#show platform health
                    %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                    Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
Lj-poll               1.00   0.02      2      1  100  500    0   0    0  1:09
GalChassisVp-review   3.00   0.29     10      3  100  500    0   0    0  11:15
S2w-JobEventSchedule 10.00   0.32     10      7  100  500    0   0    0  10:14
Stub-JobEventSchedul 10.00  12.09     10      6  100  500   14  13    9  396:35
StatValueMan Update   1.00   0.22      1      0  100  500    0   0    0  6:28
Pim-review            0.10   0.00      1      0  100  500    0   0    0  0:22
Ebm-host-review       1.00   0.00      8      0  100  500    0   0    0  0:05
Ebm-port-review       0.10   0.00      1      0  100  500    0   0    0  0:01
Protocol-aging-revie  0.20   0.00      2      0  100  500    0   0    0  0:00
Acl-Flattener  e      1.00   0.00     10      0  100  500    0   0    0  0:00
KxAclPathMan create/  1.00   0.00     10      5  100  500    0   0    0  0:39
KxAclPathMan update   2.00   0.00     10      0  100  500    0   0    0  0:00
KxAclPathMan reprogr  1.00   0.00      2      0  100  500    0   0    0  0:00
TagMan-RecreateMtegR  1.00   0.00     10      0  100  500    0   0    0  0:00
K2CpuMan Review      30.00  10.19     30     28  100  500   14  13    9  397:11
K2AccelPacketMan: Tx 10.00   2.20     20      0  100  500    2   2    1  82:06
K2AccelPacketMan: Au  0.10   0.00      0      0  100  500    0   0    0  0:00
K2AclMan-taggedFlatA  1.00   0.00     10      0  100  500    0   0    0  0:00
K2AclCamMan stale en  1.00   0.00     10      0  100  500    0   0    0  0:00
K2AclCamMan hw stats  3.00   1.04     10      5  100  500    1   1    0  39:36
K2AclCamMan kx stats  1.00   0.00     10      5  100  500    0   0    0  13:40
K2AclCamMan Audit re  1.00   0.00     10      5  100  500    0   0    0  13:10
K2AclPolicerTableMan  1.00   0.00     10      1  100  500    0   0    0  0:38
K2L2 Address Table R  2.00   0.00     12      5  100  500    0   0    0  0:00
K2L2 New Static Addr  2.00   0.00     10      1  100  500    0   0    0  0:00
K2L2 New Multicast A  2.00   0.00     10      5  100  500    0   0    0  0:01
K2L2 Dynamic Address  2.00   0.00     10      0  100  500    0   0    0  0:00
K2L2 Vlan Table Revi  2.00   0.00     12      9  100  500    0   0    0  0:01
```

```
K2 L2 Destination Ca   2.00   0.00     10      0  100  500    0   0    0  0:00
K2PortMan Review       2.00   0.72     15     11  100  500    1   1    0  37:22
Gigaport65535 Review   0.40   0.07      4      2  100  500    0   0    0  3:38
Gigaport65535 Review   0.40   0.08      4      2  100  500    0   0    0  3:39
K2Fib cam usage revi   2.00   0.00     15      0  100  500    0   0    0  0:00
K2Fib IrmFib Review    2.00   0.00     15      0  100  500    0   0    0  0:00
K2Fib Vrf Default Ro   2.00   0.00     15      0  100  500    0   0    0  0:00
K2Fib AdjRepop Revie   2.00   0.00     15      0  100  500    0   0    0  0:00
K2Fib Vrf Unpunt Rev   2.00   0.01     15      0  100  500    0   0    0  0:23
K2Fib Consistency Ch   1.00   0.00      5      2  100  500    0   0    0  29:25
K2FibAdjMan Stats Re   2.00   0.30     10      4  100  500    0   0    0  6:21
K2FibAdjMan Host Mov   2.00   0.00     10      4  100  500    0   0    0  0:00
K2FibAdjMan Adj Chan   2.00   0.00     10      0  100  500    0   0    0  0:00
K2FibMulticast Signa   2.00   0.01     10      2  100  500    0   0    0  2:04
K2FibMulticast Entry   2.00   0.00     10      7  100  500    0   0    0  0:00
K2FibMulticast Irm M   2.00   0.00     10      7  100  500    0   0    0  0:00
K2FibFastDropMan Rev   2.00   0.00      7      0  100  500    0   0    0  0:00
K2FibPbr route map r   2.00   0.06     20      5  100  500    0   0    0  16:42
K2FibPbr flat acl pr   2.00   0.07     20      2  100  500    0   0    0  3:24
K2FibPbr consolidati   2.00   0.01     10      0  100  500    0   0    0  0:24
K2FibPerVlanPuntMan    2.00   0.00     15      4  100  500    0   0    0  0:00
K2FibFlowCache flow    2.00   0.01     10      0  100  500    0   0    0  0:23
K2FibFlowCache flow    2.00   0.00     10      0  100  500    0   0    0  0:00
K2FibFlowCache adj r   2.00   0.01     10      0  100  500    0   0    0  0:20
K2FibFlowCache flow    2.00   0.00     10      0  100  500    0   0    0  0:06
K2MetStatsMan Review   2.00   0.14      5      2  100  500    0   0    0  23:40
K2FibMulticast MET S   2.00   0.00     10      0  100  500    0   0    0  0:00
K2QosDblMan Rate DBL   2.00   0.12      7      0  100  500    0   0    0  4:52
IrmFibThrottler Thro   2.00   0.01      7      0  100  500    0   0    0  0:21
K2 VlanStatsMan Revi   2.00   1.46     15      7  100  500    2   2    1  64:44
K2 Packet Memory Dia   2.00   0.00     15      8  100  500    0   1    1  45:46
K2 L2 Aging Table Re   2.00   0.12     20      3  100  500    0   0    0  7:22
RkiosPortMan Port Re   2.00   0.73     12      7  100  500    1   1    1  52:36
Rkios Module State R   4.00   0.02     40      1  100  500    0   0    0  1:28
Rkios Online Diag Re   4.00   0.02     40      0  100  500    0   0    0  1:15
RkiosIpPbr IrmPort R   2.00   0.02     10      3  100  500    0   0    0  2:44
RkiosAclMan Review     3.00   0.06     30      0  100  500    0   0    0  2:35
MatMan Review          0.50   0.00      4      0  100  500    0   0    0  0:00
Slot 3 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 3 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 4 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 4 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 5 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 5 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 6 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 6 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 7 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 7 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
EthHoleLinecardMan(1   1.66   0.04     10      0  100  500    0   0    0  1:18
EthHoleLinecardMan(2   1.66   0.02     10      0  100  500    0   0    0  1:18
EthHoleLinecardMan(6   1.66   0.17     10      6  100  500    0   0    0  6:38
                     -------------
%CPU Totals          212.80  35.63
```

Dans ce diagramme, il n'y a aucun port d'accès, mais les trois joncteurs réseau portent les VLAN 1 à 100 :

```
Switch#show platform health
                     %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                     Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
Lj-poll                1.00   0.02      2      1  100  500    0   0    0  1:09
GalChassisVp-review    3.00   0.29     10      3  100  500    0   0    0  11:15
S2w-JobEventSchedule  10.00   0.32     10      7  100  500    0   0    0  10:14
Stub-JobEventSchedul  10.00  12.09     10      6  100  500   14  13    9  396:35
```

StatValueMan Update	1.00	**0.22**	1	0	100	500	0	**0**	**0**	6:28
Pim-review	0.10	**0.00**	1	0	100	500	0	**0**	**0**	0:22
Ebm-host-review	1.00	**0.00**	8	0	100	500	0	**0**	**0**	0:05
Ebm-port-review	0.10	**0.00**	1	0	100	500	0	**0**	**0**	0:01
Protocol-aging-revie	0.20	**0.00**	2	0	100	500	0	**0**	**0**	0:00
Acl-Flattener e	1.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
KxAclPathMan create/	1.00	**0.00**	10	5	100	500	0	**0**	**0**	0:39
KxAclPathMan update	2.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
KxAclPathMan reprogr	1.00	**0.00**	2	0	100	500	0	**0**	**0**	0:00
TagMan-RecreateMtegR	1.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
K2CpuMan Review	30.00	**10.19**	30	28	100	500	14	**13**	**9**	397:11
K2AccelPacketMan: Tx	10.00	**2.20**	20	0	100	500	2	**2**	**1**	82:06
K2AccelPacketMan: Au	0.10	**0.00**	0	0	100	500	0	**0**	**0**	0:00
K2AclMan-taggedFlatA	1.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
K2AclCamMan stale en	1.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
K2AclCamMan hw stats	3.00	**1.04**	10	5	100	500	1	**1**	**0**	39:36
K2AclCamMan kx stats	1.00	**0.00**	10	5	100	500	0	**0**	**0**	13:40
K2AclCamMan Audit re	1.00	**0.00**	10	5	100	500	0	**0**	**0**	13:10
K2AclPolicerTableMan	1.00	**0.00**	10	1	100	500	0	**0**	**0**	0:38
K2L2 Address Table R	2.00	**0.00**	12	5	100	500	0	**0**	**0**	0:00
K2L2 New Static Addr	2.00	**0.00**	10	1	100	500	0	**0**	**0**	0:00
K2L2 New Multicast A	2.00	**0.00**	10	5	100	500	0	**0**	**0**	0:01
K2L2 Dynamic Address	2.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
K2L2 Vlan Table Revi	2.00	**0.00**	12	9	100	500	0	**0**	**0**	0:01
K2 L2 Destination Ca	2.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
K2PortMan Review	2.00	**0.72**	15	11	100	500	1	**1**	**0**	37:22
Gigaport65535 Review	0.40	**0.07**	4	2	100	500	0	**0**	**0**	3:38
Gigaport65535 Review	0.40	**0.08**	4	2	100	500	0	**0**	**0**	3:39
K2Fib cam usage revi	2.00	**0.00**	15	0	100	500	0	**0**	**0**	0:00
K2Fib IrmFib Review	2.00	**0.00**	15	0	100	500	0	**0**	**0**	0:00
K2Fib Vrf Default Ro	2.00	**0.00**	15	0	100	500	0	**0**	**0**	0:00
K2Fib AdjRepop Revie	2.00	**0.00**	15	0	100	500	0	**0**	**0**	0:00
K2Fib Vrf Unpunt Rev	2.00	**0.01**	15	0	100	500	0	**0**	**0**	0:23
K2Fib Consistency Ch	1.00	**0.00**	5	2	100	500	0	**0**	**0**	29:25
K2FibAdjMan Stats Re	2.00	**0.30**	10	4	100	500	0	**0**	**0**	6:21
K2FibAdjMan Host Mov	2.00	**0.00**	10	4	100	500	0	**0**	**0**	0:00
K2FibAdjMan Adj Chan	2.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
K2FibMulticast Signa	2.00	**0.01**	10	2	100	500	0	**0**	**0**	2:04
K2FibMulticast Entry	2.00	**0.00**	10	7	100	500	0	**0**	**0**	0:00
K2FibMulticast Irm M	2.00	**0.00**	10	7	100	500	0	**0**	**0**	0:00
K2FibFastDropMan Rev	2.00	**0.00**	7	0	100	500	0	**0**	**0**	0:00
K2FibPbr route map r	2.00	**0.06**	20	5	100	500	0	**0**	**0**	16:42
K2FibPbr flat acl pr	2.00	**0.07**	20	2	100	500	0	**0**	**0**	3:24
K2FibPbr consolidati	2.00	**0.01**	10	0	100	500	0	**0**	**0**	0:24
K2FibPerVlanPuntMan	2.00	**0.00**	15	4	100	500	0	**0**	**0**	0:00
K2FibFlowCache flow	2.00	**0.01**	10	0	100	500	0	**0**	**0**	0:23
K2FibFlowCache flow	2.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
K2FibFlowCache adj r	2.00	**0.01**	10	0	100	500	0	**0**	**0**	0:20
K2FibFlowCache flow	2.00	**0.00**	10	0	100	500	0	**0**	**0**	0:06
K2MetStatsMan Review	2.00	**0.14**	5	2	100	500	0	**0**	**0**	23:40
K2FibMulticast MET S	2.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
K2QosDblMan Rate DBL	2.00	**0.12**	7	0	100	500	0	**0**	**0**	4:52
IrmFibThrottler Thro	2.00	**0.01**	7	0	100	500	0	**0**	**0**	0:21
K2 VlanStatsMan Revi	2.00	**1.46**	15	7	100	500	2	**2**	**1**	64:44
K2 Packet Memory Dia	2.00	**0.00**	15	8	100	500	0	**1**	**1**	45:46
K2 L2 Aging Table Re	2.00	**0.12**	20	3	100	500	0	**0**	**0**	7:22
RkiosPortMan Port Re	2.00	**0.73**	12	7	100	500	1	**1**	**1**	52:36
Rkios Module State R	4.00	**0.02**	40	1	100	500	0	**0**	**0**	1:28
Rkios Online Diag Re	4.00	**0.02**	40	0	100	500	0	**0**	**0**	1:15
RkiosIpPbr IrmPort R	2.00	**0.02**	10	3	100	500	0	**0**	**0**	2:44
RkiosAclMan Review	3.00	**0.06**	30	0	100	500	0	**0**	**0**	2:35
MatMan Review	0.50	**0.00**	4	0	100	500	0	**0**	**0**	0:00
Slot 3 ILC Manager R	3.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00
Slot 3 ILC S2wMan Re	3.00	**0.00**	10	0	100	500	0	**0**	**0**	0:00

```
Slot 4 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 4 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 5 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 5 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 6 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 6 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 7 ILC Manager R   3.00   0.00     10      0  100  500    0   0    0  0:00
Slot 7 ILC S2wMan Re   3.00   0.00     10      0  100  500    0   0    0  0:00
EthHoleLinecardMan(1   1.66   0.04     10      0  100  500    0   0    0  1:18
EthHoleLinecardMan(2   1.66   0.02     10      0  100  500    0   0    0  1:18
EthHoleLinecardMan(6   1.66   0.17     10      6  100  500    0   0    0  6:38
                     -------------
%CPU Totals          212.80  35.63
```

Étape 1 : Vérifiez le processus de Cisco IOS avec la commande show processes cpu.

Cette section passe en revue les commandes qu'un administrateur utilise afin d'identifier le problème d'utilisation CPU élevée. Si vous émettez la commande **show processes cpu**, vous pouvez voir que deux processus, **Cat4k Mgmt LoPri** et **spanning-tree**, sont les principaux utilisateurs du CPU. Cette information vous suffit à savoir que le processus de spanning-tree utilise une importante partie des cycles CPU.

```
Switch#show processes cpu
CPU utilization for five seconds: 74%/1%; one minute: 73%; five minutes: 50%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           4       198         20  0.00%  0.00%  0.00%   0 Chunk Manager
   2           4       290         13  0.00%  0.00%  0.00%   0 Load Meter
!--- Output suppressed. 25 488 33 14787 0.00% 0.02% 0.00% 0 Per-minute Jobs 26 90656 223674 405
6.79% 6.90% 7.22% 0 Cat4k Mgmt HiPri  27      158796     59219       2681 32.55% 33.80% 21.43%
0 Cat4k Mgmt LoPri
  28          20      1693         11  0.00%  0.00%  0.00%   0 Galios Reschedul
  29           0         1          0  0.00%  0.00%  0.00%   0 IOS ACL Helper
  30           0         2          0  0.00%  0.00%  0.00%   0 NAM Manager
!--- Output suppressed. 41 0 1 0 0.00% 0.00% 0.00% 0 SFF8472 42 0 2 0 0.00% 0.00% 0.00% 0 AAA
Dictionary R  43       78564     20723       3791 32.63% 30.03% 17.35%   0 Spanning Tree
  44         112       999        112  0.00%  0.00%  0.00%   0 DTP Protocol
  45           0       147          0  0.00%  0.00%  0.00%   0 Ethchnl
```

Étape 2 : Vérifiez le processus spécifique au Catalyst 4500 à l'aide de la commande show platform health.

Afin de comprendre quel processus spécifique à une plate-forme utilise le CPU, émettez la commande **show platform health**. Cette sortie vous permet de voir que le processus **K2CpuMan Review**, une tâche de gestion des paquets liés au CPU, utilise le CPU :

```
Switch#show platform health
%CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                  Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
!--- Output suppressed. TagMan-RecreateMtegR 1.00 0.00 10 0 100 500 0 0 0 0:00 K2CpuMan Review
30.00  37.62     30     53  100  500   41  33    1  2:12
K2AccelPacketMan: Tx  10.00   4.95     20      0  100  500    5   4    0  0:36
K2AccelPacketMan: Au   0.10   0.00      0      0  100  500    0   0    0  0:00
K2AclMan-taggedFlatA   1.00   0.00     10      0  100  500    0   0    0  0:00
```

Étape 3 : Contrôlez la file d'attente du CPU qui reçoit le trafic afin d'identifier le type de trafic lié au CPU.

Émettez la commande **show platform cpu packet statistics** afin de contrôler quelle file d'attente CPU reçoit le paquet lié au CPU. La sortie de cette section montre que la file d'attente de contrôle reçoit beaucoup de paquets. Utilisez les informations du tableau 1 et la conclusion à laquelle vous

avez abouti lors de l'étape 1. Vous pouvez déterminer que le traitement des BPDU est à l'origine des paquets que le CPU traite et de l'utilisation élevée du CPU.

```
Switch#show platform cpu packet statistics
!--- Output suppressed. Total packet queues 16 Packets Received by Packet Queue Queue Total 5
sec avg 1 min avg 5 min avg 1 hour avg ---------------------- --------------- --------- --------
- --------- ---------- Esmp 202760 196 173 128 28 Control                          388623
2121      1740       598         16

Packets Dropped by Packet Queue

Queue                   Total           5 sec avg 1 min avg 5 min avg 1 hour avg
---------------------- --------------- --------- --------- --------- ----------
Control                           17918         0        19        24          3
```

Étape 4 : Identifiez la cause principale.

Émettez la commande **show spanning-tree summary**. Vous pouvez vérifier si la réception des BPDU est due au nombre élevé d'instances de port de spanning-tree. La sortie identifie clairement la cause principale :

```
Switch#show spanning-tree summary
Switch is in pvst mode
Root bridge for: none
Extended system ID           is enabled
Portfast Default             is disabled
PortFast BPDU Guard Default  is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default            is disabled
EtherChannel misconfig guard is enabled
UplinkFast                   is disabled
BackboneFast                 is disabled
Configured Pathcost method used is short
!--- Output suppressed. Name Blocking Listening Learning Forwarding STP Active -----------------
----- -------- --------- -------- ---------- ---------- 2994 vlans                    0
0        0       5999       5999
```

Il existe un grand nombre de VLAN avec la configuration de mode PVST+. Afin de résoudre ce problème, changez le mode STP en Multiple Spanning Tree (MST). Dans certains cas, le nombre d'instances STP est élevé car un grand nombre de VLAN sont transférés sur tous les ports de jonction. Dans ce cas, supprimez manuellement les VLAN qui ne sont pas nécessaires à la liaison afin de diminuer le nombre de ports STP actifs bien au-dessous de la valeur recommandée.

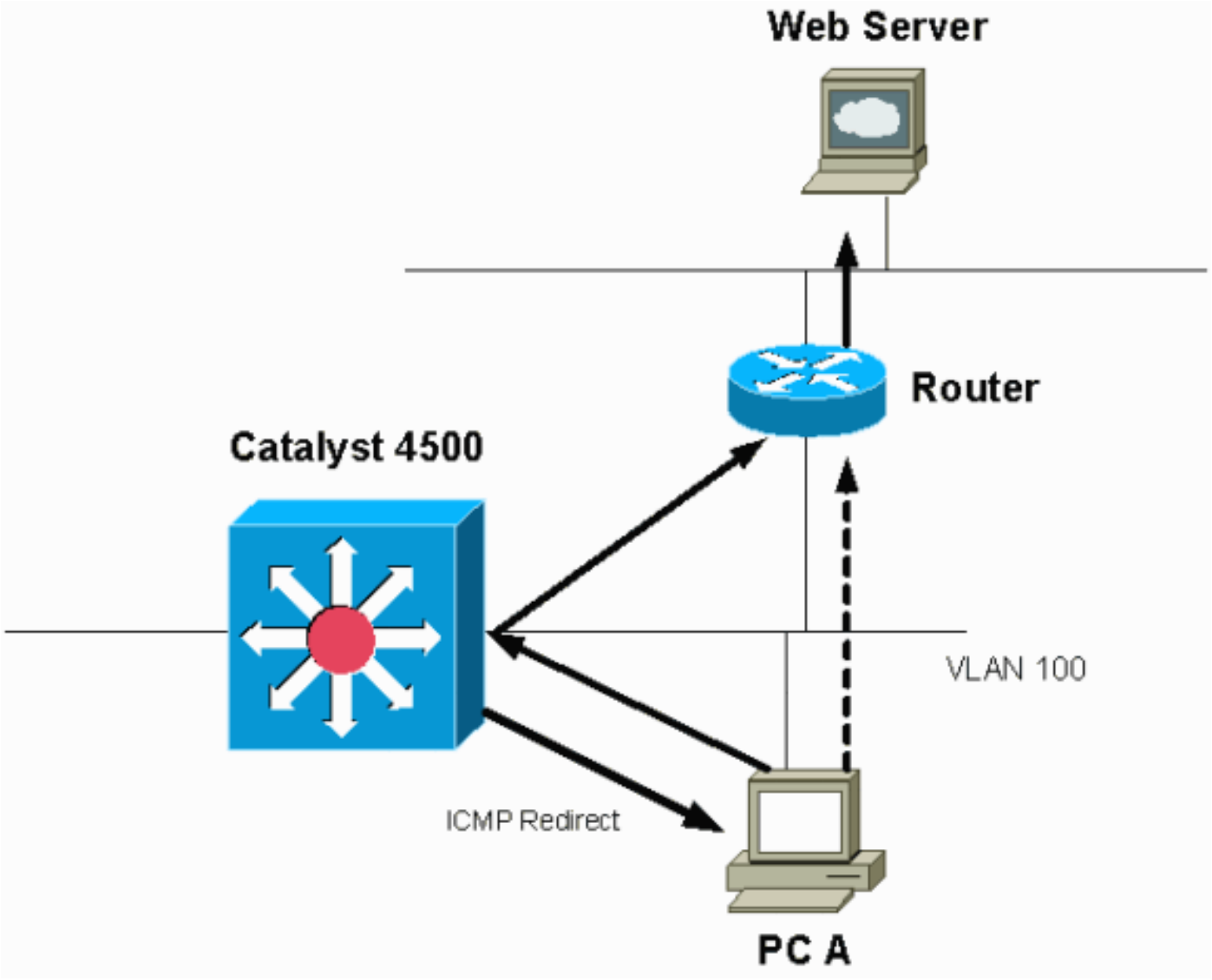
Conseil : Assurez-vous que vous ne configurez pas les ports de téléphone IP comme ports de jonction. Il s'agit d'une erreur de configuration courante. Configurez les ports de téléphone IP avec une configuration VLAN voix. Cette configuration crée une pseudo liaison, mais n'exige pas que vous supprimiez manuellement les VLAN inutiles. Pour plus d'informations sur la configuration des ports de voix, référez-vous au guide de configuration logicielle Configurer les interfaces voix. Les téléphones IP non-Cisco ne prennent pas en charge cette configuration VLAN voix ou VLAN auxiliaire. Vous devez supprimer manuellement les ports liés à des téléphones IP non-Cisco.

Redirections ICMP ; Routage de paquets sur la même interface

Le routage de paquets sur la même interface, ou l'entrée et la sortie de trafic sur la même interface L3, peut entraîner une redirection ICMP par le commutateur. Si le commutateur sait que le prochain périphérique de saut vers la destination finale est dans le même sous-réseau que le périphérique émetteur, il génère une redirection ICMP vers la source. Les messages de redirection indiquent à la source d'envoyer le paquet directement au prochain périphérique de

saut. Le message indique que le prochain périphérique de saut a un meilleur itinéraire de destination, un itinéraire comprenant un saut de moins que ce commutateur.

Dans le diagramme de cette section, le PC A communique avec le serveur Web. La passerelle par défaut du PC A indique l'adresse IP de l'interface VLAN 100. Cependant, le prochain routeur de saut qui permet au Catalyst 4500 d'atteindre sa destination est dans le même sous-réseau que le PC A. Dans ce cas, passer directement par le « routeur » est plus rapide. Le Catalyst 4500 envoie un message de redirection ICMP au PC A. Le message demande au PC A d'envoyer les paquets destinés au serveur Web par l'intermédiaire de routeur, plutôt que par le Catalyst 4500. Cependant, dans la plupart des cas, les périphériques ne répondent pas à la redirection ICMP. Cette absence de réponse entraine l'utilisation par le Catalyst 4500 d'un grand nombre de cycles CPU pour la génération de ces redirections d'ICMP pour tous les paquets que Catalyst transfert par l'intermédiaire de la même interface que les paquets d'entrée.



La redirection ICMP est activée par défaut. Pour la désactiver, utilisez la commande **no ip icmp redirects**. Émettez la commande sous l'interface SVI ou L3 pertinente.

Remarque: Puisque **ip icmp redirects** est une commande par défaut, elle n'est pas visible dans la sortie de commande **show running-configuration**.

[Étape 1 : Vérifiez le processus de Cisco IOS avec la commande show processes cpu.]

Émettez la commande **show processes cpu**. Vous pouvez voir que deux processus, **Cat4k Mgmt LoPri** et **IP Input**, sont les principaux utilisateurs du CPU. Cette information vous suffit à savoir que le traitement des paquets IP utilise une grande partie du CPU.

```
Switch#show processes cpu
CPU utilization for five seconds: 38%/1%; one minute: 32%; five minutes: 32%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           0        63          0  0.00%  0.00%  0.00%   0 Chunk Manager
   2          60     50074          1  0.00%  0.00%  0.00%   0 Load Meter
   3           0         1          0  0.00%  0.00%  0.00%   0 Deferred Events
!--- Output suppressed. 27 524 250268 2 0.00% 0.00% 0.00% 0 TTY Background 28 816 254843 3 0.00%
0.00% 0.00% 0 Per-Second Jobs 29 101100 5053 20007 0.00% 0.01% 0.00% 0 Per-minute Jobs 30
26057260 26720902 975 5.81% 6.78% 5.76% 0 Cat4k Mgmt HiPri 31    19482908  29413060        662
19.64% 18.20% 20.48%   0 Cat4k Mgmt LoPri
!--- Output suppressed. 35 60 902 0 0.00% 0.00% 0.00% 0 DHCP Snooping 36   504625304 645491491
781 72.40% 72.63% 73.82%   0 IP Input
```

Étape 2 : Vérifiez le processus spécifique au Catalyst 4500 à l'aide de la commande show platform health.

La sortie de la commande **show platform health** confirme le pourcentage de CPU utilisé pour traiter les paquets liés au CPU.

```
Switch#show platform health
%CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                     Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
--- Output suppressed. TagMan-RecreateMtegR 1.00 0.00 10 0 100 500 0 0 0 0:00 K2CpuMan Review
330.00 19.18 150 79 25 500 20 19 18 5794:08 K2AccelPacketMan: Tx 10.00 4.95 20 0 100 500 5 4 0
0:36 K2AccelPacketMan: Au 0.10 0.00 0 0 100 500 0 0 0 0:00 K2AclMan-taggedFlatA 1.00 0.00 10 0
100 500 0 0 0 0:00
```

Étape 3 : Contrôlez la file d'attente du CPU qui reçoit le trafic afin d'identifier le type de trafic lié au CPU.

Émettez la commande **show platform cpu packet statistics** afin de contrôler quelle file d'attente CPU reçoit le paquet lié au CPU. Vous pouvez voir que la file d'attente L3 Fwd Low reçoit énormément de trafic.

```
Switch#show platform cpu packet statistics
!--- Output suppressed. Packets Received by Packet Queue Queue Total 5 sec avg 1 min avg 5 min
avg 1 hour avg ---------------------- --------------- --------- --------- --------- ----------
Esmp 48613268 38 39 38 39 Control 142166648 74 74 73 73 Host Learning 1845568 2 2 2 2 L3 Fwd
High 17 0 0 0 0 L3 Fwd Medium 2626 0 0 0 0 L3 Fwd Low                  4717094264      3841
3879      3873       3547
L2 Fwd Medium                          1         0         0         0          0
L3 Rx High                        257147         0         0         0          0
L3 Rx Low                        5325772        10        19        13          7
RPF Failure                          155         0         0         0          0
ACL fwd(snooping)               65604591        53        54        54         53
ACL log, unreach                11013420         9         8         8          8
```

Étape 4 : Identifiez la cause principale.

Dans ce cas, utilisez CPU SPAN afin de déterminer le trafic qui atteint le CPU. Pour des informations concernant CPU SPAN, voyez l'outil 1 : Surveiller le trafic CPU avec le logiciel SPAN-Cisco IOS Version 12.1(19)EW et ultérieure de ce document. Effectuez une analyse du trafic et une configuration à l'aide de la commande **show running-configuration**. Dans ce cas, un paquet est routé par la même interface, qui mène au problème de redirection ICMP pour chaque paquet. Cette cause principale est l'une des raisons courantes d'une utilisation CPU élevée sur le

Catalyst 4500.

Vous pouvez vous attendre à ce que le périphérique d'accès réagisse à la redirection ICMP que le Catalyst 4500 envoie et modifie le prochain saut pour la destination. Cependant, tous les périphériques ne répondent pas à une redirection ICMP. Si le périphérique ne répond pas, le Catalyst 4500 doit envoyer des redirections pour chaque paquet que reçoit le commutateur d'un périphérique émetteur. Ces redirections peuvent utiliser beaucoup de ressources CPU. La solution est de désactiver la redirection ICMP. Émettez la commande **no ip redirects** sous les interfaces.

Ce scénario peut se produire lorsque vous avez également configuré des adresses IP secondaires. Lorsque vous activez les adresses IP secondaires, la redirection d'IP est automatiquement désactivée. Assurez-vous de ne pas activer manuellement les redirections d'IP.

Comme l'indique cette section intitulée Redirections ICMP ; routage de paquets sur la même interface, la plupart des périphériques ne répondent pas aux redirections ICMP. Par conséquent, de manière générale, désactivez cette fonctionnalité.

Routage IPX ou AppleTalk

Le Catalyst 4500 prend en charge le routage IPX et AppleTalk par l'intermédiaire d'un chemin de transfert logiciel uniquement. Avec configuration de tels protocoles, une utilisation CPU plus élevée est normale.

Remarque: La commutation du trafic IPX et AppleTalk dans le même VLAN ne nécessite pas de commutation de processus. Seuls les paquets qui doivent être routés nécessitent un transfert de chemin logiciel.

Étape 1 : Vérifiez le processus de Cisco IOS avec la commande show processes cpu.

Émettez la commande **show processes cpu** afin de vérifier quel processus de Cisco IOS utilise le CPU. Dans cette sortie de commande, notez que le processus supérieur est **Cat4k Mgmt LoPri** :

```
witch#show processes cpu
CPU utilization for five seconds: 87%/10%; one minute: 86%; five minutes: 87%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           4        53         75  0.00%  0.00%  0.00%   0 Chunk Manager
!--- Output suppressed. 25 8008 1329154 6 0.00% 0.00% 0.00% 0 Per-Second Jobs 26 413128 38493
10732 0.00% 0.02% 0.00% 0 Per-minute Jobs 27 148288424 354390017 418 2.60% 2.42% 2.77% 0 Cat4k
Mgmt HiPri 28   285796820 720618753        396 50.15% 59.72% 61.31%   0 Cat4k Mgmt LoPri
```

Étape 2 : Vérifiez le processus spécifique au Catalyst 4500 à l'aide de la commande show platform health.

La sortie de la commande **show platform health** confirme le pourcentage de CPU utilisé pour traiter les paquets liés au CPU.

```
Switch#show platform health
                    %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                    Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
!--- Output suppressed. TagMan-RecreateMtegR 1.00 0.00 10 4 100 500 0 0 0 0:00 K2CpuMan Review
30.00  27.39     30     53  100  500   42  47   42  4841:
K2AccelPacketMan: Tx  10.00   8.03     20      0  100  500   21  29   26  270:4
```

Étape 3 : Contrôlez la file d'attente du CPU qui reçoit le trafic afin d'identifier le type de trafic lié au CPU.

Afin de déterminer le type de trafic qui atteint le CPU, émettez la commande **show platform cpu packet statistics**.

```
Switch#show platform cpu packet statistics
!--- Output suppressed. Packets Received by Packet Queue Queue Total 5 sec avg 1 min avg 5 min
avg 1 hour avg --------------------- -------------- --------- --------- --------- ----------
Esmp 48613268 38 39 38 39 Control 142166648 74 74 73 73 Host Learning 1845568 2 2 2 2 L3 Fwd
High 17 0 0 0 0 L3 Fwd Medium 2626 0 0 0 0 L3 Fwd Low 1582414 1 1 1 1 L2 Fwd Medium 1 0 0 0 0 L2
Fwd Low                    576905398      1837      1697      1938       1515
L3 Rx High                    257147         0         0         0          0
L3 Rx Low                    5325772        10        19        13          7
RPF Failure                      155         0         0         0          0
ACL fwd(snooping)            65604591        53        54        54         53
ACL log, unreach             11013420         9         8         8          8
```

Étape 4 : Identifiez la cause principale.

Puisque l'administrateur a configuré le routage IPX ou AppleTalk, l'identification de la cause principale devrait être simple. Mais pour le confirmer, utilisez SPAN sur le trafic CPU et assurez-vous que le trafic que vous voyez est le trafic attendu. Pour des informations concernant CPU SPAN, voyez l'outil 1 : Surveiller le trafic CPU avec le logiciel SPAN-Cisco IOS Version 12.1(19)EW et ultérieure de ce document.

Dans ce cas, l'administrateur doit mettre à jour la spécification de base du CPU avec la valeur actuelle. Le CPU du Catalyst 4500 se comporte comme prévu lorsqu'il traite les paquets commutés par logiciel.

Apprentissage d'hôte

Le Catalyst 4500 apprend les adresses MAC de plusieurs hôtes si l'adresse MAC n'est pas déjà dans la table des adresses MAC. Le moteur de commutation transfert une copie du paquet avec la nouvelle adresse MAC au CPU.

Toutes les interfaces VLAN (couche 3) utilisent l'adresse matérielle de base de châssis comme adresse MAC. En conséquence, la table d'adresses MAC ne comporte aucune entrée et les paquets destinés à ces interfaces VLAN ne sont pas envoyés au CPU pour traitement.

S'il le nombre de nouvelles adresses MAC est trop important pour que le commutateur les apprenne, l'utilisation CPU peut augmenter.

Étape 1 : Vérifiez le processus de Cisco IOS avec la commande show processes cpu.

Émettez la commande **show processes cpu** afin de vérifier quel processus de Cisco IOS utilise le CPU. Dans cette sortie de commande, notez que le processus supérieur est **Cat4k Mgmt LoPri** :

```
Switch#show processes cpu
CPU utilization for five seconds: 89%/1%; one minute: 74%; five minutes: 71%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           4        53         75  0.00%  0.00%  0.00%   0 Chunk Manager
!--- Output suppressed. 25 8008 1329154 6 0.00% 0.00% 0.00% 0 Per-Second Jobs 26 413128 38493
10732 0.00% 0.02% 0.00% 0 Per-minute Jobs 27 148288424 354390017 418 26.47% 10.28% 10.11% 0
Cat4k Mgmt HiPri 28   285796820 720618753        396 52.71% 56.79% 55.70%   0 Cat4k Mgmt LoPri
```

Étape 2 : Vérifiez le processus spécifique au Catalyst 4500 à l'aide de la commande show platform health.

La sortie de la commande **show platform health** confirme le pourcentage de CPU utilisé pour traiter les paquets liés au CPU.

```
Switch#show platform health
                    %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                    Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
!--- Output suppressed. TagMan-RecreateMtegR 1.00 0.00 10 4 100 500 0 0 0 0:00 K2CpuMan Review
30.00  46.88     30     47  100  500   30  29   21  265:01
K2AccelPacketMan: Tx  10.00   8.03     20      0  100  500   21  29   26  270:4
```

Étape 3 : Contrôlez la file d'attente du CPU qui reçoit le trafic afin d'identifier le type de trafic lié au CPU.

Afin de déterminer le type de trafic qui atteint le CPU, émettez la commande **show platform cpu packet statistics**.

```
Switch#show platform cpu packet statistics
!--- Output suppressed. Packets Received by Packet Queue Queue Total 5 sec avg 1 min avg 5 min
avg 1 hour avg ---------------------- -------------- --------- --------- --------- ----------
Esmp 48613268 38 39 38 39 Control 142166648 74 74 73 73 Host Learning                 1845568
1328      1808      1393       1309
L3 Fwd High                          17         0         0         0          0
L3 Fwd Medium                      2626         0         0         0          0
L3 Fwd Low                      1582414         1         1         1          1
L2 Fwd Medium                         1         0         0         0          0
L2 Fwd Low                    576905398        37         7         8          5
L3 Rx High                       257147         0         0         0          0
L3 Rx Low                       5325772        10        19        13          7
RPF Failure                         155         0         0         0          0
ACL fwd(snooping)              65604591        53        54        54         53
ACL log, unreach               11013420         9         8         8          8
```

Étape 4 : Identifiez la cause principale.

La sortie de la commande **show platform health** vous indique que le CPU voit beaucoup de nouvelles adresses MAC. Cette situation est souvent le résultat de l'instabilité de topologie du réseau. Par exemple, si la topologie du spanning-tree change, le commutateur génère des notifications de modification de topologie (TCN). L'émission des TCN réduit le temps de vieillissement à 15 secondes en mode PVST+. Les entrées d'adresses MAC sont éliminées si les adresses ne sont pas réapprises dans le délai prévu. Dans le cas de RSTP (Rapid STP) (IEEE 802.1w) ou de MST (IEEE 802.1s), les entrées expirent immédiatement si le TCN provient d'un autre commutateur. Cette expiration fait que les adresses MAC doivent être à nouveau apprises. Il ne s'agit pas d'un problème grave si les modifications de topologie sont rares. Mais un lien instable, un commutateur défectueux ou des ports hôtes non autorisés pour PortFast peuvent entraîner un nombre excessif de modifications de topologie. Ceci peut entraîner le vidage de nombreuses tables MAC et donc nécessiter un nouvel apprentissage. L'étape suivante dans l'identification de la cause principale consiste à dépanner le réseau. Le commutateur fonctionne comme prévu et envoie les paquets au CPU pour l'apprentissage des adresses d'hôtes. Identifiez et réparez le périphérique défectueux qui entraîne une génération excessive de TCN.

Votre réseau peut contenir de nombreux périphériques qui envoient le trafic par à-coups, ce qui fait expirer les adresse MAC qui doivent ensuite être apprises à nouveau par le commutateur. Dans ce cas, augmentez le délai de vieillissement de la table d'adresses MAC afin d'améliorer la situation. Avec un délai de vieillissement plus long, les commutateurs retiennent les adresses MAC dans la table plus longtemps avant d'expirer.

Attention : Modifiez cette expiration après y avoir bien réfléchi. Cette modification peut entraîner

un trou noir dans le trafic si votre réseau comporte des périphériques mobiles.

Manque de ressources matérielles (TCAM) pour la sécurité de liste de contrôle d'accès

Catalyst 4500 programme les ACL configurées à l'aide du TCAM Cisco. TCAM permet l'application des ACL dans le chemin de transfert matériel. Il n'y a aucune incidence sur la performance du commutateur, avec ou sans ACL dans le chemin de transfert. La performance est constante quelle que soit la taille de l'ACL car la performance des recherches ACL est à plein débit. Cependant, TCAM n'est pas une ressource inépuisable. Par conséquent, si vous configurez un nombre excessif d'entrées ACL, vous dépasserez la capacité TCAM. Le tableau 3 montre le nombre de ressources TCAM disponibles sur chacun des moteurs de superviseur et commutateurs Catalyst 4500.

Tableau 3 - Capacité TCAM sur les moteurs de superviseur et commutateurs Catalyst 4500

Produit	Fonction TCAM (par direction)	QoS TCAM (par direction)
Supervisor Engine II+/II+TS	8 192 entrées avec 1 024masques	8 192 entrées avec 1 024masques
Supervisor Engine III/IV/V et Catalyst 4948	16 384 entrées avec 2 048 masques	16 384 entrées avec 2 048 masques
Supervisor Engine V-10GE et Catalyst 4948-10GE	16 384 entrées avec 16 384 masques	16 384 entrées avec 16 384 masques

Le commutateur utilise la caractéristique TCAM afin de programmer la sécurité ACL, comme RACL et VLAN ACL (VACL). Le commutateur utilise également TCAM pour les fonctions de sécurité comme la protection de la source IP (IPSG) pour les ACL dynamiques. Le commutateur utilise QoS TCAM afin de programmer la classification et les ACL de l'applicateur de stratégies.

Lorsque le Catalyst 4500 vient à manquer de ressources TCAM lors de la programmation d'une sécurité ACL, une application partielle de l'ACL se produit par l'intermédiaire du chemin logiciel. Les paquets qui atteignent ces ACE sont traités dans le logiciel, ce qui entraîne une utilisation élevée du CPU. L'ACL est programmée de haut en bas. En d'autres termes, si l'ACL ne s'insère pas dans le TCAM, l'ACE de la partie inférieure de l'ACL n'est vraisemblablement pas programmée dans le TCAM.

Ce message d'avertissement apparaît lorsqu'un débordement TCAM se produit :

```
Switch#show platform cpu packet statistics
!--- Output suppressed. Packets Received by Packet Queue Queue Total 5 sec avg 1 min avg 5 min
avg 1 hour avg ---------------------- -------------- --------- --------- --------- ----------
Esmp 48613268 38 39 38 39 Control 142166648 74 74 73 73 Host Learning                 1845568
1328      1808      1393       1309
L3 Fwd High                          17         0         0         0          0
L3 Fwd Medium                      2626         0         0         0          0
L3 Fwd Low                      1582414         1         1         1          1
L2 Fwd Medium                         1         0         0         0          0
L2 Fwd Low                    576905398        37         7         8          5
L3 Rx High                       257147         0         0         0          0
```

```
L3 Rx Low                      5325772        10        19        13         7
RPF Failure                        155         0         0         0         0
ACL fwd(snooping)              65604591        53        54        54        53
ACL log, unreach               11013420         9         8         8         8
```

Vous pouvez voir ce message d'erreur dans la sortie de commande **show logging**. Ce message indique de façon certaine qu'un traitement logiciel aura lieu et, par conséquent, qu'il peut y avoir utilisation CPU élevée.

Remarque: Si vous modifiez une grande ACL, vous pouvez voir ce message brièvement avant que l'ACL modifiée ne soit programmé de nouveau dans le TCAM.

Étape 1 : Vérifiez le processus de Cisco IOS avec la commande show processes cpu.

Émettez la commande **show processes cpu**. Vous pouvez voir que l'utilisation CPU est élevée car le processus **Cat4k Mgmt LoPri** utilise la plupart des cycles CPU.

```
Switch#show processes cpu
CPU utilization for five seconds: 99%/0%; one minute: 99%; five minutes: 99%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           0        11          0  0.00%  0.00%  0.00%   0 Chunk Manager
   2        9716    632814         15  0.00%  0.00%  0.00%   0 Load Meter
   3         780       302       2582  0.00%  0.00%  0.00%   0 SpanTree Helper
!--- Output suppressed. 23 18208 3154201 5 0.00% 0.00% 0.00% 0 TTY Background 24 37208 3942818 9
0.00% 0.00% 0.00% 0 Per-Second Jobs 25 1046448 110711 9452 0.00% 0.03% 0.00% 0 Per-minute Jobs
26 175803612 339500656 517 4.12% 4.31% 4.48% 0 Cat4k Mgmt HiPri   27   835809548 339138782
2464 86.81% 89.20% 89.76%   0 Cat4k Mgmt LoPri
  28       28668   2058810         13  0.00%  0.00%  0.00%   0 Galios Reschedul
```

Étape 2 : Vérifiez le processus spécifique au Catalyst 4500 à l'aide de la commande show platform health.

Émettez la commande **show platform health**. Vous pouvez voir que **K2CpuMan Review**, une tâche de gestion des paquets liés au CPU, utilise le CPU.

```
Switch#show platform health
%CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                    Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
Lj-poll               1.00   0.01      2      0  100  500    0   0    0  13:45
GalChassisVp-review    3.00   0.20     10     16  100  500    0   0    0  88:44
S2w-JobEventSchedule  10.00   0.57     10      7  100  500    1   0    0  404:22
Stub-JobEventSchedul  10.00   0.00     10      0  100  500    0   0    0  0:00
StatValueMan Update    1.00   0.09      1      0  100  500    0   0    0  91:33
Pim-review             0.10   0.00      1      0  100  500    0   0    0  4:46
Ebm-host-review        1.00   0.00      8      4  100  500    0   0    0  14:01
Ebm-port-review        0.10   0.00      1      0  100  500    0   0    0  0:20
Protocol-aging-revie   0.20   0.00      2      0  100  500    0   0    0  0:01
Acl-Flattener          1.00   0.00     10      5  100  500    0   0    0  0:04
KxAclPathMan create/   1.00   0.00     10      5  100  500    0   0    0  0:21
KxAclPathMan update    2.00   0.00     10      6  100  500    0   0    0  0:05
KxAclPathMan reprogr   1.00   0.00      2      1  100  500    0   0    0  0:00
TagMan-InformMtegRev   1.00   0.00      5      0  100  500    0   0    0  0:00
TagMan-RecreateMtegR   1.00   0.00     10     14  100  500    0   0    0  0:18
K2CpuMan Review       30.00  91.31     30     92  100  500  128 119   84  13039:02
K2AccelPacketMan: Tx  10.00   2.30     20      0  100  500    2   2    2  1345:30
K2AccelPacketMan: Au   0.10   0.00      0      0  100  500    0   0    0  0:00
```

Étape 3 : Contrôlez la file d'attente du CPU qui reçoit le trafic afin d'identifier le type de trafic lié au CPU.

Vous devez comprendre quelle file d'attente CPU et donc quel type de traffic atteint la file d'attente CPU. Émettez la commande **show platform cpu packet statistics**. Vous pouvez voir que la file d'attente ACL sw processing reçoit un nombre élevé de paquets. Par conséquent, le débordement TCAM est la cause de ce problème d'utilisation CPU élevée.

```
Switch#show platform cpu packet statistics
!--- Output suppressed. Packets Received by Packet Queue Queue Total 5 sec avg 1 min avg 5 min
avg 1 hour avg ---------------------- -------------- --------- --------- --------- ----------
Control 57902635 22 16 12 3 Host Learning 464678 0 0 0 0 L3 Fwd Low 623229 0 0 0 0 L2 Fwd Low
11267182 7 4 6 1 L3 Rx High 508 0 0 0 0 L3 Rx Low 1275695 10 1 0 0 ACL fwd(snooping) 2645752 0 0
0 0 ACL log, unreach 51443268 9 4 5 5 ACL sw processing           842889240      1453      1532
1267       1179

Packets Dropped by Packet Queue

Queue                   Total           5 sec avg 1 min avg 5 min avg 1 hour avg
---------------------- -------------- --------- --------- --------- ----------
L2 Fwd Low                        3270         0         0         0          0
ACL sw processing                12636         0         0         0          0
```

Étape 4 : Résolvez le problème.

Dans l'étape 3, vous avez déterminé la cause principale dans ce scénario. Supprimez l'ACL qui a entraîné le débordement ou réduisez au minimum l'ACL pour éviter le débordement. Passez également en revue les directives concernant la configuration Configurer la sécurité du réseau avec les ACL afin d'optimiser la configuration et la programmation d'ACL dans le matériel.

Mot clé log dans la liste de contrôle d'accès

Le Catalyst 4500 prend en charge la journalisation de détails de paquets qui atteignent n'importe quelle entrée ACL. Toutefois, une journalisation excessive peut entraîner une utilisation CPU élevée. Évitez l'utilisation des **mots-clés de journal**, sauf pendant l'étape de découverte du trafic. Pendant l'étape de découverte du trafic, vous identifiez le trafic qui traverse votre network pour lequel vous n'avez pas explicitement configuré d'ACE. N'utilisez pas le **mot-clé de journal** afin de recueillir des statistiques. Dans le Logiciel Cisco IOS Version 12.1(13)EW et ultérieure, les **messages du journal** sont limités en débit. Si vous utilisez des **messages du journal** afin de compter le nombre de paquets qui correspondent à l'ACL, le compte n'est pas précis. Au lieu de cela, utilisez la commande **show access-list** afin d'obtenir des statistiques précises. L'identification de cette cause principale est plus facile parce qu'un examen de la configuration ou des **messages du journal** peut indiquer l'utilisation de la fonctionnalité de journalisation ACL.

Étape 1 : Vérifiez le processus de Cisco IOS avec la commande show processes cpu.

Émettez la commande **show processes cpu** afin de vérifier quel processus de Cisco IOS utilise le CPU. Dans cette sortie de commande, notez que le processus supérieur est **Cat4k Mgmt LoPri** :

```
Switch#show processes cpu
CPU utilization for five seconds: 99%/0%; one minute: 99%; five minutes: 99%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           0        11          0  0.00%  0.00%  0.00%   0 Chunk Manager
   2        9716    632814         15  0.00%  0.00%  0.00%   0 Load Meter
!--- Output suppressed. 26 175803612 339500656 517 4.12% 4.31% 4.48% 0 Cat4k Mgmt HiPri   27
835809548 339138782       2464 86.81% 89.20% 89.76%   0 Cat4k Mgmt LoPri
  28       28668   2058810         13  0.00%  0.00%  0.00%   0 Galios Reschedul
```

Étape 2 : Vérifiez le processus spécifique au Catalyst 4500 à l'aide de la commande show

platform health.

Contrôlez le processus spécifique à une plate-forme qui utilise le CPU. Émettez la commande **show platform health**. Dans la sortie, notez que **K2CpuMan Review process** utilise la plupart des cycles CPU. Cette activité indique que le CPU est occupé, car il traite les paquets qui lui sont destinés.

```
Switch#show platform health
                      %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                    Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
Lj-poll               1.00   0.01      2      0  100  500    0   0    0  13:45
GalChassisVp-review    3.00   0.20     10     16  100  500    0   0    0  88:44
S2w-JobEventSchedule  10.00   0.57     10      7  100  500    1   0    0  404:22
Stub-JobEventSchedul  10.00   0.00     10      0  100  500    0   0    0  0:00
StatValueMan Update    1.00   0.09      1      0  100  500    0   0    0  91:33
Pim-review             0.10   0.00      1      0  100  500    0   0    0  4:46
Ebm-host-review        1.00   0.00      8      4  100  500    0   0    0  14:01
Ebm-port-review        0.10   0.00      1      0  100  500    0   0    0  0:20
Protocol-aging-revie   0.20   0.00      2      0  100  500    0   0    0  0:01
Acl-Flattener          1.00   0.00     10      5  100  500    0   0    0  0:04
KxAclPathMan create/   1.00   0.00     10      5  100  500    0   0    0  0:21
KxAclPathMan update    2.00   0.00     10      6  100  500    0   0    0  0:05
KxAclPathMan reprogr   1.00   0.00      2      1  100  500    0   0    0  0:00
TagMan-InformMtegRev   1.00   0.00      5      0  100  500    0   0    0  0:00
TagMan-RecreateMtegR   1.00   0.00     10     14  100  500    0   0    0  0:18
K2CpuMan Review       30.00  91.31     30     92  100  500  128 119   84  13039:02
K2AccelPacketMan: Tx  10.00   2.30     20      0  100  500    2   2    2  1345:30
K2AccelPacketMan: Au   0.10   0.00      0      0  100  500    0   0    0  0:00
```

Étape 3 : Contrôlez la file d'attente du CPU qui reçoit le trafic afin d'identifier le type de trafic lié au CPU.

Afin de déterminer le type de trafic qui atteint le CPU, émettez la commande **show platform cpu packet statistics**. Dans cette sortie de commande, vous pouvez voir que la réception de paquets est due au **mot-clé de journal d'**ACL :

```
Switch#show platform cpu packet statistics
!--- Output suppressed. Total packet queues 16 Packets Received by Packet Queue Queue Total 5
sec avg 1 min avg 5 min avg 1 hour avg ---------------- -------------------- --------- --------
- --------- ---------- Control 1198701435 35 35 34 35 Host Learning 874391 0 0 0 0 L3 Fwd High
428 0 0 0 0 L3 Fwd Medium 12745 0 0 0 0 L3 Fwd Low 2420401 0 0 0 0 L2 Fwd High 26855 0 0 0 0 L2
Fwd Medium 116587 0 0 0 0 L2 Fwd Low 317829151 53 41 31 31 L3 Rx High 2371 0 0 0 0 L3 Rx Low
32333361 7 1 2 0 RPF Failure 4127 0 0 0 0 ACL fwd (snooping) 107743299 4 4 4 4 ACL log, unreach
1209056404      1987      2125      2139       2089

Packets Dropped by Packet Queue

Queue             Total                5 sec avg 1 min avg 5 min avg 1 hour avg
----------------- -------------------- --------- --------- --------- ----------
ACL log, unreach             193094788       509       362       437        394
```

Étape 4 : Résolvez le problème.

Dans l'étape 3, vous avez déterminé la cause principale dans ce scénario. Afin d'éviter ce problème, supprimez le **mot-clé de journal** des ACL. Dans le logiciel Cisco IOS Version 12.1(13)EW1 et version ultérieure, les paquets sont limités en débit de sorte que l'utilisation du CPU ne soit pas trop élevée. Utilisez les compteurs de listes d'accès afin de garder une trace des consultations ACL. Vous pouvez voir les compteurs de listes d'accès dans la sortie de commande **show access-list acl_id**.

Boucles de transfert de la couche 2

Les boucles de transfert de la couche 2 peuvent être provoquées par une mauvaise mise en œuvre du protocole Spanning Tree protocol (STP) et divers problèmes qui peuvent affecter STP.

Étape 1 : Vérifiez le processus de Cisco IOS avec la commande show processes cpu

Cette section passe en revue les commandes qu'un administrateur utilise afin d'identifier le problème d'utilisation CPU élevée. Si vous émettez la commande **show processes cpu**, vous pouvez voir que deux processus, **Cat4k Mgmt LoPri** et **spanning-tree**, sont les principaux utilisateurs du CPU. Cette information vous suffit à savoir que le processus de spanning-tree utilise une importante partie des cycles CPU.

```
Switch#show processes cpu
CPU utilization for five seconds: 74%/1%; one minute: 73%; five minutes: 50%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           4       198         20  0.00%  0.00%  0.00%   0 Chunk Manager
   2           4       290         13  0.00%  0.00%  0.00%   0 Load Meter
!--- Output suppressed. 25 488 33 14787 0.00% 0.02% 0.00% 0 Per-minute Jobs 26 90656 223674 405
6.79% 6.90% 7.22% 0 Cat4k Mgmt HiPri  27      158796     59219       2681 32.55% 33.80% 21.43%
0 Cat4k Mgmt LoPri
  28          20      1693         11  0.00%  0.00%  0.00%   0 Galios Reschedul
  29           0         1          0  0.00%  0.00%  0.00%   0 IOS ACL Helper
  30           0         2          0  0.00%  0.00%  0.00%   0 NAM Manager
!--- Output suppressed. 41 0 1 0 0.00% 0.00% 0.00% 0 SFF8472 42 0 2 0 0.00% 0.00% 0.00% 0 AAA
Dictionary R  43       78564     20723       3791 32.63% 30.03% 17.35%   0 Spanning Tree
  44         112       999        112  0.00%  0.00%  0.00%   0 DTP Protocol
  45           0       147          0  0.00%  0.00%  0.00%   0 Ethchnl
```

Étape 2 : Vérifiez le processus spécifique à Catalyst 4500 à l'aide de la commande show platform health

Afin de comprendre quel processus spécifique à une plate-forme utilise le CPU, émettez la commande **show platform health**. Cette sortie vous permet de voir que le processus **K2CpuMan Review**, une tâche de gestion des paquets liés au CPU, utilise le CPU :

```
Switch#show platform health
%CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                    Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
!--- Output suppressed. TagMan-RecreateMtegR 1.00 0.00 10 0 100 500 0 0 0 0:00 K2CpuMan Review
30.00  37.62     30     53  100  500   41  33    1  2:12
K2AccelPacketMan: Tx  10.00   4.95     20      0  100  500    5   4    0  0:36
K2AccelPacketMan: Au   0.10   0.00      0      0  100  500    0   0    0  0:00
K2AclMan-taggedFlatA   1.00   0.00     10      0  100  500    0   0    0  0:00
```

Étape 3 : Contrôlez la file d'attente du CPU qui reçoit le trafic afin d'identifier le type de trafic lié au CPU

Émettez la commande **show platform cpu packet statistics** afin de contrôler quelle file d'attente CPU reçoit le paquet lié au CPU. La sortie de cette section montre que la file d'attente de contrôle reçoit beaucoup de paquets. Utilisez les informations du tableau 1 et la conclusion à laquelle vous avez abouti lors de l'étape 1. Vous pouvez déterminer que le traitement des BPDU est à l'origine des paquets que le CPU traite et de l'utilisation élevée du CPU.

```
Switch#show platform cpu packet statistics
!--- Output suppressed. Total packet queues 16 Packets Received by Packet Queue Queue Total 5
sec avg 1 min avg 5 min avg 1 hour avg ---------------------- -------------- --------- --------
```

```
- --------- ---------- Esmp 202760 196 173 128 28 Control                      388623
2121      1740       598         16

Packets Dropped by Packet Queue

Queue                  Total          5 sec avg 1 min avg 5 min avg 1 hour avg
---------------------- -------------- --------- --------- --------- ----------
Control                          17918         0        19        24          3
```

Étape 4 : Identifiez la cause principale et résolvez le problème

Généralement, vous pouvez effectuer ces étapes pour procéder au dépannage (selon la situation, certaines étapes ne sont pas nécessaires) :

1. Identifiez la boucle.
2. Découvrez la portée de la boucle.
3. Cassez la boucle.
4. Corrigez la cause de la boucle.
5. Restaurez la redondance.

Chacune des étapes est expliquée en détails dans Dépannage des boucles de transfert - Dépannage de STP sur des commutateurs Catalyst exécutant le logiciel Cisco IOS System.

Étape 5 : Mettez en œuvre les fonctionnalités STP avancées

- **BDPU Guard -** Protège STP contre les périphériques réseau non autorisés connectés aux ports portfast activés. Pour plus d'informations, référez-vous à Amélioration de Spanning Tree PortFast BPDU Guard.
- **Loop Guard -** Augmente la stabilité des réseaux de couche 2. Pour plus d'informations, référez-vous à Amélioration du protocole Spanning Tree à l'aide des fonctionnalités de protection contre les boucles et de détection des différences de temps de propagation des BPDU.
- **Root Guard -** Applique le placement du pont racine dans le réseau. Référez-vous à Perfectionnement de la protection de la racine du protocole Spanning Tree pour plus d'informations.
- **UDLD** — Détecte les liens unidirectionnels et empêche des boucles de transfert. Pour plus d'informations, référez-vous à Comprendre et configurer la fonctionnalité protocole UDLD (UniDirectional Link Detection).

Autres causes d'une utilisation CPU élevée

Voici quelques autres causes connues d'utilisation CPU élevée :

- Instabilités excessives du lien
- Pics d'utilisation CPU dus au contrôle de cohérence FIB
- Utilisation élevée du CPU dans le processus **K2FibAdjMan Host Move**
- Utilisation du CPU élevé dans le processus de **RkiosPortMan Port Review**
- Utilisation du CPU élevé une fois connecté à un téléphone IP à l'utilisation des ports de joncteur réseau
- Utilisation CPU élevée avec RSPAN et les paquets de contrôle de la couche 3
- Pic pendant la programmation d'une ACL de grande tailleLa pointe dans l'utilisation CPU se

produit pendant l'application ou la suppression d'une ACL de grande taille d'une interface.

Instabilités excessives du lien

Le Catalyst 4500 fait état d'une utilisation élevée du CPU lorsqu'un ou plusieurs des liens attachés deviennent trop instables. Cette situation se produit dans des versions du logiciel Cisco IOS antérieures au logiciel Cisco IOS Version 12.2(20)EWA.

Étape 1 : Vérifiez le processus de Cisco IOS avec la commande show processes cpu.

Émettez la commande **show processes cpu** afin de vérifier quel processus de Cisco IOS utilise le CPU. Dans cette sortie de commande, notez que le processus supérieur est **Cat4k Mgmt LoPri** :

```
Switch#show processes cpu
CPU utilization for five seconds: 96%/0%; one minute: 76%; five minutes: 68%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           0         4          0  0.00%  0.00%  0.00%   0 Chunk Manager
   2        9840    463370         21  0.00%  0.00%  0.00%   0 Load Meter
   3           0         2          0  0.00%  0.00%  0.00%   0 SNMP Timers
!--- Output suppressed. 27 232385144 530644966 437 13.98% 12.65% 12.16% 0 Cat4k Mgmt HiPri   28
564756724 156627753       3605 64.74% 60.71% 54.75%   0 Cat4k Mgmt LoPri
  29        9716   1806301          5  0.00%  0.00%  0.00%   0 Galios Reschedul
```

Étape 2 : Vérifiez le processus spécifique au Catalyst 4500 à l'aide de la commande show platform health.

La sortie de la commande **show platform health** indique que le processus **KxAclPathMan Create** utilise la plupart des ressources CPU. Ce processus est dédié à la création d'un chemin interne.

```
Switch#show platform health
                     %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                     Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
Lj-poll                1.00   0.03      2      0  100  500    0   0    0  9:49
GalChassisVp-review    3.00   1.11     10     62  100  500    0   0    0  37:39
S2w-JobEventSchedule  10.00   2.85     10      8  100  500    2   2    2  90:00
Stub-JobEventSchedul  10.00   5.27     10      9  100  500    4   4    4  186:2
Pim-review             0.10   0.00      1      0  100  500    0   0    0  2:51
Ebm-host-review        1.00   0.00      8      4  100  500    0   0    0  8:06
Ebm-port-review        0.10   0.00      1      0  100  500    0   0    0  0:14
Protocol-aging-revie   0.20   0.00      2      0  100  500    0   0    0  0:00
Acl-Flattener          1.00   0.00     10      5  100  500    0   0    0  0:00
KxAclPathMan create/   1.00  69.11     10      5  100  500   42  53   22  715:0
KxAclPathMan update    2.00   0.76     10      6  100  500    0   0    0  86:00
KxAclPathMan reprogr   1.00   0.00      2      1  100  500    0   0    0  0:00
TagMan-InformMtegRev   1.00   0.00      5      0  100  500    0   0    0  0:00
TagMan-RecreateMtegR   1.00   0.00     10    227  100  500    0   0    0  0:00
K2CpuMan Review       30.00   8.05     30     57  100  500    6   5    5  215:0
K2AccelPacketMan: Tx  10.00   6.86     20      0  100  500    5   5    4  78:42
```

Étape 3 : Identifiez la cause principale.

Activez la journalisation pour les messages d'établissement/interruption de liaison. Cette journalisation n'est pas activée par défaut. Son activation vous aide à déterminer très rapidement les liens posant problème. Émettez la commande **logging event link-status** sous toutes les interfaces. Vous pouvez utiliser la commande **interface range** afin de lancer la commande sur une série d'interfaces, comme le montre l'exemple ci-dessous :

```
Switch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#interface range gigabitethernet 5/1 - 48
Switch(config-if-range)#logging event link-status
Switch(config--if-range)#end
Switch#show logging
!--- Output suppressed. 3w5d: %LINK-3-UPDOWN: Interface GigabitEthernet5/24, changed state to
down 3w5d: %LINK-3-UPDOWN: Interface GigabitEthernet5/24, changed state to up 3w5d: %LINK-3-
UPDOWN: Interface GigabitEthernet5/24, changed state to down 3w5d: %LINK-3-UPDOWN: Interface
GigabitEthernet5/24, changed state to up 3w5d: %LINK-3-UPDOWN: Interface GigabitEthernet5/24,
changed state to down 3w5d: %LINK-3-UPDOWN: Interface GigabitEthernet5/24, changed state to up
```

Après avoir identifié l'interface défectueuse ou instable, éteignez-la afin de résoudre le problème d'utilisation CPU élevée. Le logiciel Cisco IOS Version 12.2(20)EWA et ultérieure ont amélioré le comportement du Catalyst 4500 en cas de lien instable. Par conséquent, l'incidence sur le CPU n'est plus aussi importante qu'avant l'amélioration. Rappelez-vous que ce processus est un processus d'arrière-plan. L'utilisation CPU élevée due à ce problème n'entraîne pas d' effets indésirables sur les commutateurs Catalyst 4500.

Pics d'utilisation CPU dus au contrôle de cohérence FIB

Les commutateurs Catalyst 4500 peuvent faire état de pics momentanés d'utilisation CPU pendant un contrôle de cohérence d'une table FIB. La table FIB est la table de transfert L3 créée par le processus CEF. Le contrôle de cohérence maintient la cohérence entre la table FIB du logiciel Cisco IOS et les entrées matérielles. Cette cohérence assure que les paquets ne sont pas routés de manière incorrecte. Le contrôle a lieu toutes les 2 secondes et fonctionne comme processus en arrière-plan non prioritaire. Ce processus se comporte normalement et n'interfère pas avec d'autres processus ou paquets hautement prioritaires.

La sortie de la commande **show platform health** montre que **K2Fib Consistency Ch** utilise la plus grande partie du CPU.

Remarque: L'utilisation moyenne du CPU pour ce processus est insignifiante au-delà d'une minute ou d'une heure, ce qui confirme que le contrôle est un examen périodique court. Ce processus en arrière-plan utilise uniquement des cycles CPU inactifs.

```
Switch#show platform health
                     %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                     Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
Lj-poll                1.00   0.02      2      1  100  500    0   0    0  1:09
GalChassisVp-review    3.00   0.29     10      3  100  500    0   0    0  11:15
!--- Output suppressed. K2Fib cam usage revi 2.00 0.00 15 0 100 500 0 0 0 0:00 K2Fib IrmFib
Review 2.00 0.00 15 0 100 500 0 0 0 0:00 K2Fib Vrf Default Ro 2.00 0.00 15 0 100 500 0 0 0 0:00
K2Fib AdjRepop Revie 2.00 0.00 15 0 100 500 0 0 0 0:00 K2Fib Vrf Unpunt Rev 2.00 0.01 15 0 100
500 0 0 0 0:23 K2Fib Consistency Ch 1.00 60.40 5 2 100 500 0   0    0 100:23
K2FibAdjMan Stats Re   2.00   0.30     10      4  100  500    0   0    0  6:21
K2FibAdjMan Host Mov   2.00   0.00     10      4  100  500    0   0    0  0:00
K2FibAdjMan Adj Chan   2.00   0.00     10      0  100  500    0   0    0  0:00
K2FibMulticast Signa   2.00   0.01     10      2  100  500    0   0    0  2:04
```

Utilisation élevée du CPU dans le processus K2FibAdjMan Host Move

Le Catalyst 4500 peut présenter une utilisation CPU élevée dans le processus **K2FibAdjMan Host Move**. Cette utilisation élevée apparaît dans la sortie de la commande **show platform health**. De nombreuses adresses MAC expirent fréquemment ou sont apprises sur de nouveaux ports, ce qui entraîne cette utilisation élevée du CPU. La valeur par défaut du temps de vieillissement de la table d'adresses MAC est de 5 minutes (300 secondes). La solution à ce problème consiste à augmenter le temps de vieillissement des adresses MAC ou vous pouvez concevoir le réseau de

façon à éviter un nombre élevé de mouvements d'adresses MAC. Le logiciel Cisco IOS Version 12.2(18)EW et ultérieure ont amélioré le comportement de ce processus afin d'utiliser moins de CPU. Référez-vous à l'ID bogue Cisco CSCed15021 (clients enregistrés uniquement).

```
Switch#show platform health
                     %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                     Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
Lj-poll                1.00   0.02      2      1  100  500    0   0    0  1:09
GalChassisVp-review    3.00   0.29     10      3  100  500    0   0    0  11:15
S2w-JobEventSchedule  10.00   0.32     10      7  100  500    0   0    0  10:14
!--- Output suppressed. K2FibAdjMan Stats Re 2.00 0.30 10 4 100 500 0 0 0 6:21 K2FibAdjMan Host
Mov   2.00  18.68     10      4  100  500   25  29   28  2134:39
K2FibAdjMan Adj Chan   2.00   0.00     10      0  100  500    0   0    0  0:00
K2FibMulticast Signa   2.00   0.01     10      2  100  500    0   0    0  2:04
K2FibMulticast Entry   2.00   0.00     10      7  100  500    0   0    0  0:00
```

Vous pouvez modifier le temps de vieillissement maximum d'une adresse MAC dans le mode de configuration globale. La syntaxe de la commande est **mac-address-table aging-time seconds** pour un routeur et **mac-address-table aging-time seconds [vlan vlan-id]** pour un commutateur Catalyst. Pour plus d'informations, référez-vous au Guide de référence des commandes de services de commutation de Cisco IOS.

Utilisation CPU élevée dans le processus RkiosPortMan Port Review

Le Catalyst 4500 peut présenter une utilisation élevée du CPU dans le processus **RkiosPortMan Port Review** dans la sortie de la commande **show platform health** dans le logiciel Cisco IOS Version 12.2(25)EWA et 12.2(25)EWA1. L'ID bogue Cisco CSCeh08768 (clients enregistrés uniquement) entraîne une utilisation CPU élevée, que le logiciel Cisco IOS Version 12.2(25)EWA2 résout. Ce processus est un processus en arrière-plan et n'affecte pas la stabilité des commutateurs Catalyst 4500.

```
Switch#show platform health
                     %CPU   %CPU    RunTimeMax   Priority  Average %CPU  Total
                     Target Actual Target Actual   Fg   Bg 5Sec Min Hour  CPU
Lj-poll                1.00   0.02      2      1  100  500    0   0    0  1:09
GalChassisVp-review    3.00   0.29     10      3  100  500    0   0    0  11:15
S2w-JobEventSchedule  10.00   0.32     10      7  100  500    0   0    0  10:14
!--- Output suppressed. K2 Packet Memory Dia 2.00 0.00 15 8 100 500 0 1 1 45:46 K2 L2 Aging
Table Re 2.00 0.12 20 3 100 500 0 0 0 7:22 RkiosPortMan Port Re   2.00  87.92     12      7  100
500   99  99   89  1052:36
Rkios Module State R   4.00   0.02     40      1  100  500    0   0    0  1:28
Rkios Online Diag Re   4.00   0.02     40      0  100  500    0   0    0  1:15
```

Utilisation CPU élevée une fois connecté à un téléphone IP avec l'utilisation des ports de jonction

Si un port est configuré pour l'option VLAN voix et VLAN accès, le port sert de port d'accès multi-VLAN. L'avantage est que seuls les VLAN configurés pour les options de voix et d'accès VLAN sont liés.

Les VLAN qui sont liés au téléphone augmentent le nombre d'instances STP. Le commutateur gère les instances STP. La gestion de l'augmentation des instances STP augmente également l'utilisation CPU du STP.

La liaison de tous les VLAN entraîne également une diffusion, un Multicast et un trafic unicast inconnu vers le lien téléphonique.

```
Switch#show processes cpu
CPU utilization for five seconds: 69%/0%; one minute: 72%; five minutes: 73%
```

```
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           4       165         24  0.00%  0.00%  0.00%   0 Chunk Manager
   2       29012    739091         39  0.00%  0.00%  0.00%   0 Load Meter
   3       67080     13762       4874  0.00%  0.00%  0.00%   0 SpanTree Helper
   4           0         1          0  0.00%  0.00%  0.00%   0 Deferred Events
   5           0         2          0  0.00%  0.00%  0.00%   0 IpSecMibTopN
   6     4980144    570766       8725  0.00%  0.09%  0.11%   0 Check heaps
  26   539173952 530982442       1015 13.09% 13.05% 13.20%   0 Cat4k Mgmt HiPri
  27   716335120 180543127       3967 17.61% 18.19% 18.41%   0 Cat4k Mgmt LoPri
  33     1073728     61623      17424  0.00%  0.03%  0.00%   0 Per-minute Jobs
  34  1366717824 231584970       5901 38.99% 38.90% 38.92%   0 Spanning Tree
  35     2218424  18349158        120  0.00%  0.03%  0.02%   0 DTP Protocol
  36        5160    369525         13  0.00%  0.00%  0.00%   0 Ethchnl
  37      271016   2308022        117  0.00%  0.00%  0.00%   0 VLAN Manager
  38      958084   3965585        241  0.00%  0.01%  0.01%   0 UDLD
  39        1436     51011         28  0.00%  0.00%  0.00%   0 DHCP Snooping
  40         780     61658         12  0.00%  0.00%  0.00%   0 Port-Security
  41     1355308  12210934        110  0.00%  0.01%  0.00%   0 IP Input
```

Utilisation CPU élevée avec RSPAN et les paquets de contrôle de la couche 3

Les paquets de contrôle de la couche 3 capturés avec RSPAN sont destinés au CPU plutôt qu'uniquement à l'interface de destination du RSPAN, ce qui entraîne une utilisation CPU élevée. Les paquets de contrôle L3 sont capturés par des entrées CAM statiques, puis transférées à l'action CPU. Les entrées CAM statiques sont communes à tous les VLAN. Afin d'éviter une utilisation CPU excessive, utilisez la fonctionnalité Per-VLAN Control Traffic Intercept, disponible dans le logiciel Cisco IOS Version 12.2(37)SG et ultérieure.

```
Switch#show processes cpu
CPU utilization for five seconds: 69%/0%; one minute: 72%; five minutes: 73%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1           4       165         24  0.00%  0.00%  0.00%   0 Chunk Manager
   2       29012    739091         39  0.00%  0.00%  0.00%   0 Load Meter
   3       67080     13762       4874  0.00%  0.00%  0.00%   0 SpanTree Helper
   4           0         1          0  0.00%  0.00%  0.00%   0 Deferred Events
   5           0         2          0  0.00%  0.00%  0.00%   0 IpSecMibTopN
   6     4980144    570766       8725  0.00%  0.09%  0.11%   0 Check heaps
  26   539173952 530982442       1015 13.09% 13.05% 13.20%   0 Cat4k Mgmt HiPri
  27   716335120 180543127       3967 17.61% 18.19% 18.41%   0 Cat4k Mgmt LoPri
  33     1073728     61623      17424  0.00%  0.03%  0.00%   0 Per-minute Jobs
  34  1366717824 231584970       5901 38.99% 38.90% 38.92%   0 Spanning Tree
  35     2218424  18349158        120  0.00%  0.03%  0.02%   0 DTP Protocol
  36        5160    369525         13  0.00%  0.00%  0.00%   0 Ethchnl
  37      271016   2308022        117  0.00%  0.00%  0.00%   0 VLAN Manager
  38      958084   3965585        241  0.00%  0.01%  0.01%   0 UDLD
  39        1436     51011         28  0.00%  0.00%  0.00%   0 DHCP Snooping
  40         780     61658         12  0.00%  0.00%  0.00%   0 Port-Security
  41     1355308  12210934        110  0.00%  0.01%  0.00%   0 IP Input
```

Les ACL statiques sont installées en haut de la fonctionnalité d'entrée TCAM afin de capturer des paquets de contrôle destinés à des adresses Multicast IP connues dans la plage 224.0.0.*. Les ACL statiques sont installées au moment du démarrage et apparaissent avant toute ACL configurée par l'utilisateur. Les ACL statiques sont toujours consultées en premier et arrêtent le trafic de contrôle vers le CPU sur tous les VLAN.

La fonctionnalité Per-VLAN control traffic intercept fournit un mode géré de chemin par VLAN sélectif de capture du trafic de contrôle. Les entrées CAM statiques correspondantes dans la fonctionnalité TCAM d'entrée sont invalidées dans le nouveau mode. Des paquets de contrôle sont capturés par l'ACL spécifique à une fonction attachée aux VLAN sur lesquels les fonctionnalités de snooping et de routage sont activées. Aucun ACL spécifique à une fonctionnalité n'est attaché au VLAN du RSPAN. Par conséquent, aucun des paquets de contrôle

de la couche 3 provenant du VLAN du RSPAN n'est transféré au CPU.

Outils de dépannage d'analyse du trafic destiné au CPU

Comme l'a montré ce document, le trafic destiné au CPU constitue l'une des principales causes d'une utilisation CPU élevée sur les Catalyst 4500. Le trafic destiné au CPU peut être intentionnel en raison de la configuration, ou involontaire en raison d'une mauvaise configuration ou d'une attaque de déni de service. Le CPU dispose d' un mécanisme QoS incorporé afin d'empêcher tous les effets indésirables sur le réseau causés par ce trafic. Cependant, identifiez la cause principale du trafic lié au CPU et éliminez le trafic s'il se révèle indésirable.

Outil 1 : Surveiller le trafic CPU avec le logiciel SPAN-Cisco IOS Version 12.1(19)EW ou ultérieure

Le Catalyst 4500 permet la surveillance du trafic lié au CPU, d'entrée ou de sortie, à l'aide de la fonction standard SPAN. L'interface de destination se connecte un outil de surveillance des paquets ou à un ordinateur portable d'administrateur qui exécute le logiciel renifleur de paquet. Cet outil aide à analyser rapidement et précisément le trafic que traite le CPU. Cet outil permet de surveiller les files d'attente individuelles qui sont liées au moteur de paquet du CPU.

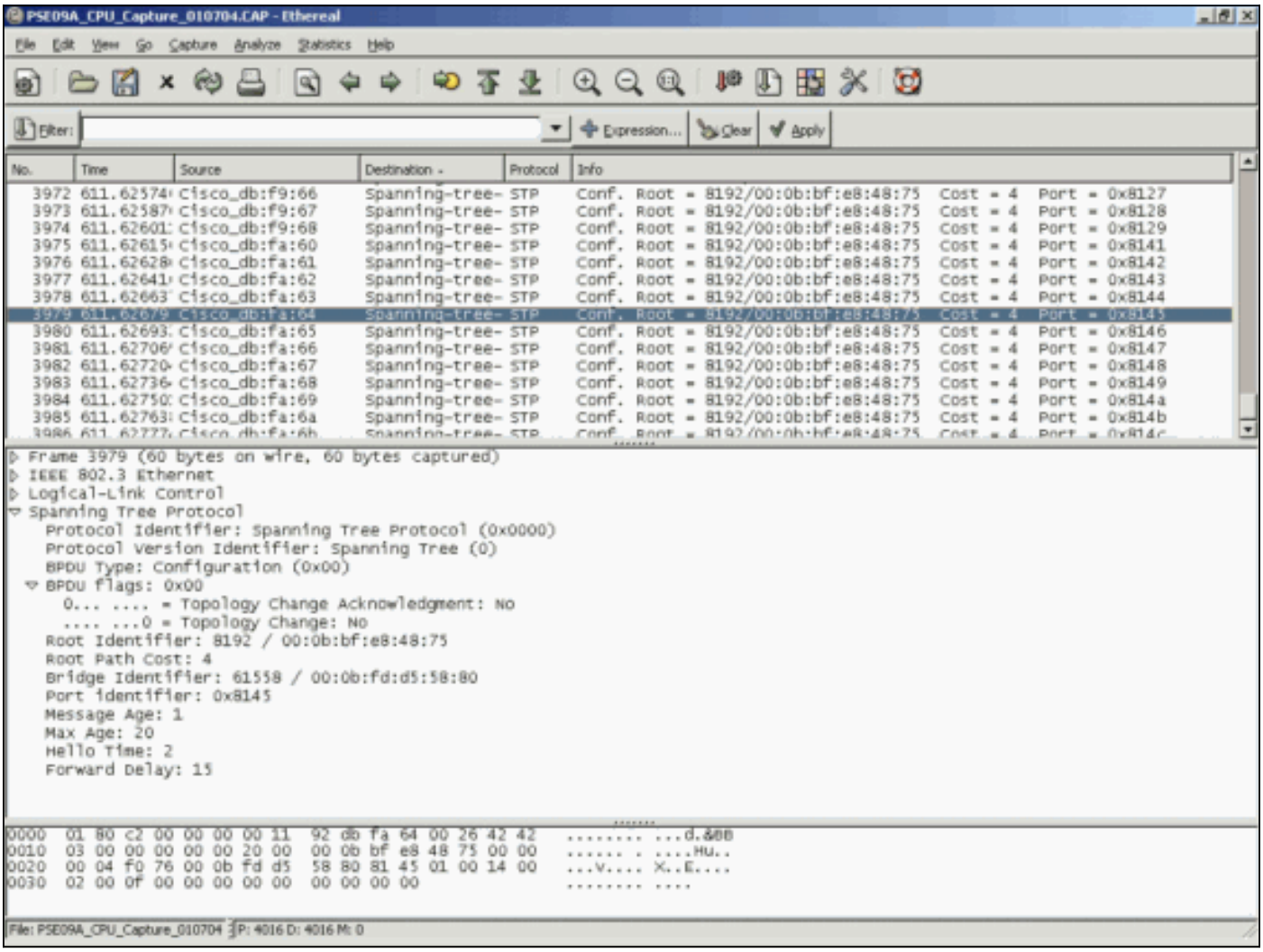
Remarque: Le moteur de commutation dispose de 32 files d'attente pour le trafic CPU et le moteur du CPU de 16 files d'attente.

```
Switch(config)#monitor session 1 source cpu ?
  both   Monitor received and transmitted traffic
  queue  SPAN source CPU queue
  rx     Monitor received traffic only
  tx     Monitor transmitted traffic only
  <cr>
Switch(config)#monitor session 1 source cpu queue ?
  <1-32>          SPAN source CPU queue numbers
  acl             Input and output ACL [13-20]
  adj-same-if     Packets routed to the incoming interface [7]
  all             All queues [1-32]
  bridged         L2/bridged packets [29-32]
  control-packet  Layer 2 Control Packets [5]
  mtu-exceeded    Output interface MTU exceeded [9]
  nfl             Packets sent to CPU by netflow (unused) [8]
  routed          L3/routed packets [21-28]
  rpf-failure     Multicast RPF Failures [6]
  span            SPAN to CPU (unused) [11]
  unknown-sa      Packets with missing source address [10]
Switch(config)#monitor session 1 source cpu queue all rx
Switch(config)#monitor session 1 destination interface gigabitethernet 1/3
Switch(config)#end
4w6d: %SYS-5-CONFIG_I: Configured from console by console

Switch#show monitor session 1
Session 1
---------
Type              : Local Session
Source Ports      :
    RX Only       : CPU
Destination Ports : Gi1/3
    Encapsulation : Native
          Ingress : Disabled
         Learning : Disabled
```

Si vous connectez un PC qui exécute un programme de renifleur, vous pouvez analyser rapidement le trafic. Dans la sortie qui apparaît dans la fenêtre de cette section, vous pouvez voir que l'utilisation élevée du CPU est due à un nombre excessif des BPDU de STP.

Remarque: La présence de BPDU de STP dans le renifleur du CPU est normale. Mais si vous en voyez plus que prévu, vous pouvez avoir dépassé les limites recommandées pour votre moteur de superviseur. Pour plus d'informations, voyez la section Un nombre élevé d'instances de port spanning-tree de ce document.



Outil 2 : Renifleur incorporé CPU — Version du logiciel Cisco IOS 12.2(20)EW et plus tard

Le Catalyst 4500 fournit un renifleur et décodeur de CPU incorporé pour identifier rapidement le trafic qui atteint le CPU. Vous pouvez activer cette fonctionnalité avec la commande **debug**, comme le montre l'exemple de cette section. Cette fonctionnalité applique une mémoire tampon circulaire qui peut retenir 1 024 paquets simultanément. Lorsque de nouveaux paquets arrivent, ils écrasent les plus anciens. Cette fonctionnalité peut être utilisée en toute sécurité lors du dépannage de problèmes d'utilisation CPU élevée.

```
Switch#debug platform packet all receive buffer
platform packet debugging is on
Switch#show platform cpu packet buffered
Total Received Packets Buffered: 36
-----------------------------------
Index 0:
```

```
7 days 23:6:32:37214 - RxVlan: 99, RxPort: Gi4/48
Priority: Crucial, Tag: Dot1Q Tag, Event: Control Packet, Flags: 0x40, Size: 68
Eth: Src 00-0F-F7-AC-EE-4F Dst 01-00-0C-CC-CC-CD Type/Len 0x0032
Remaining data:
 0: 0xAA 0xAA 0x3  0x0  0x0  0xC  0x1  0xB  0x0  0x0
10: 0x0  0x0  0x0  0x80 0x0  0x0  0x2  0x16 0x63 0x28
20: 0x62 0x0  0x0  0x0  0x0  0x80 0x0  0x0  0x2  0x16
30: 0x63 0x28 0x62 0x80 0xF0 0x0  0x0  0x14 0x0  0x2
40: 0x0  0xF  0x0  0x0  0x0  0x0  0x0  0x2  0x0  0x63
Index 1:
7 days 23:6:33:180863 - RxVlan: 1, RxPort: Gi4/48
Priority: Crucial, Tag: Dot1Q Tag, Event: Control Packet, Flags: 0x40, Size: 68
Eth: Src 00-0F-F7-AC-EE-4F Dst 01-00-0C-CC-CC-CD Type/Len 0x0032
Remaining data:
 0: 0xAA 0xAA 0x3  0x0  0x0  0xC  0x1  0xB  0x0  0x0
10: 0x0  0x0  0x0  0x80 0x0  0x0  0x2  0x16 0x63 0x28
20: 0x62 0x0  0x0  0x0  0x0  0x80 0x0  0x0  0x2  0x16
30: 0x63 0x28 0x62 0x80 0xF0 0x0  0x0  0x14 0x0  0x2
40: 0x0  0xF  0x0  0x0  0x0  0x0  0x0  0x2  0x0  0x63
```

Remarque: Lorsque vous émettez une commande **debug,** l'utilisation CPU est toujours de presque 100 %. Il est normal d'avoir une utilisation CPU élevée lorsque vous émettez une commande **debug**.

Outil 3 : Identifier l'interface qui envoie le trafic au CPU - Cisco IOS Version 12.2(20)SW ou ultérieure

Catalyst 4500 fournit un autre outil utile pour identifier les interfaces supérieures qui envoient du trafic/des paquets pour traitement par le CPU. Cet outil vous aide à identifier rapidement un périphérique qui envoie un nombre élevé de diffusions ou d'autres attaques par déni de service au CPU. Cette fonctionnalité est également sûre pour une utilisation lors du dépannage de problèmes d'utilisation CPU élevée.

```
Switch#debug platform packet all count
platform packet debugging is on
Switch#show platform cpu packet statistics
!--- Output suppressed. Packets Transmitted from CPU per Output Interface Interface Total 5 sec
avg 1 min avg 5 min avg 1 hour avg ---------------------- -------------- --------- --------- --
------- ---------- Gi4/47 1150 1 5 10 0 Gi4/48 50 1 0 0 0 Packets Received at CPU per Input
Interface

Interface              Total           5 sec avg 1 min avg 5 min avg 1 hour avg
---------------------- --------------- --------- --------- --------- ----------
Gi4/47                         23130         5         10        50          20
Gi4/48                            50         1          0         0           0
```

Remarque: Lorsque vous émettez une commande **debug,** l'utilisation CPU est toujours de presque 100 %. Il est normal d'avoir une utilisation CPU élevée lorsque vous émettez une commande **debug**.

Résumé

Les commutateurs Catalyst 4500 gèrent un débit élevé de transfert de paquets de la version d'IP 4 (ipv4) dans le matériel. Certaines des fonctionnalités ou des exceptions peuvent entraîner le transfert de certains paquets par l'intermédiaire du chemin de traitement du CPU. Le Catalyst 4500 utilise un mécanisme QoS sophistiqué pour gérer les paquets liés au CPU. Ce mécanisme assure la fiabilité et la stabilité des commutateurs tout en maximisant le CPU pour le transfert logiciel de paquets. Le logiciel Cisco IOS Version 12.2(25)EWA2 et ultérieure fournit des

améliorations supplémentaires pour la gestion de paquets/processus ainsi que le comptage. Catalyst 4500 dispose également de commandes suffisantes et d'outils puissants pour faciliter l'identification de la cause principale des scénarios d'utilisation élevée du CPU. Mais, dans la plupart des cas, l'utilisation élevée du CPU sur un Catalyst 4500 n'est pas une cause d'instabilité du réseau ni un sujet d'inquiétude.

Informations connexes

- Utilisation du processeur sur les commutateurs Catalyst 4500/4000, 2948G, 2980G et 4912G qui exécutent le logiciel CatOS
- Pages de support pour les produits LAN
- Page de support sur la commutation LAN
- Support et documentation techniques - Cisco Systems