

# 對Catalyst 4500交換機上的CPU使用率過高進行故障排除

## 目錄

---

### [簡介](#)

#### [必要條件](#)

[需求](#)

[採用元件](#)

[慣例](#)

#### [背景資訊](#)

##### [瞭解Catalyst 4500 CPU資料包處理架構](#)

##### [確定Catalyst 4500上CPU使用率高的原因](#)

[稽核CPU使用情況](#)

[瞭解Catalyst 4500交換機上的show processes cpu命令](#)

[瞭解Catalyst 4500交換機上的show platform health命令](#)

##### [排除常見的CPU使用率較高問題](#)

###### [進程交換資料包造成的CPU使用率高](#)

[大量的生成樹埠例項](#)

[第2步：使用show platform health命令檢查Catalyst 4500特定的進程。](#)

[第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。](#)

[第4步：確定根本原因。](#)

###### [ICMP重定向：在同一介面上路由資料包](#)

[第1步：使用show processes cpu命令檢查Cisco IOS進程。](#)

[第2步：使用show platform health命令檢查Catalyst 4500特定的進程。](#)

[第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。](#)

[第4步：確定根本原因。](#)

###### [IPX或AppleTalk路由](#)

[第1步：使用show processes cpu命令檢查Cisco IOS進程。](#)

[第2步：使用show platform health命令檢查Catalyst 4500特定的進程。](#)

[第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。](#)

[第4步：確定根本原因。](#)

###### [主機學習](#)

[第1步：使用show processes cpu命令檢查Cisco IOS進程。](#)

[第2步：使用show platform health命令檢查Catalyst 4500特定的進程。](#)

[第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。](#)

[第4步：確定根本原因。](#)

###### [安全ACL的硬體資源不足\(TCAM\)](#)

[第1步：使用show processes cpu命令檢查Cisco IOS進程。](#)

[第2步：使用show platform health命令檢查Catalyst 4500特定的進程。](#)

[第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。](#)

[第4步：解決問題。](#)

###### [ACL中的log關鍵字](#)

[第1步：使用show processes cpu命令檢查Cisco IOS進程。](#)

[第2步：使用show platform health命令檢查Catalyst 4500特定的進程。](#)

[第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。](#)

[第4步：解決問題。](#)

## [第2層轉發環路](#)

[第1步：使用show processes cpu命令檢查Cisco IOS進程](#)

[第2步：使用show platform health命令檢查Catalyst 4500特定的進程](#)

[第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別](#)

[第4步：確定根本原因並解決問題](#)

[第5步：實施高級STP功能](#)

## [CPU使用率高的其他原因](#)

### [過度連結翻動](#)

[第1步：使用show processes cpu命令檢查Cisco IOS進程。](#)

[第2步：使用show platform health命令檢查Catalyst 4500特定的進程。](#)

[第3步：確定根本原因。](#)

[FIB一致性檢查導致的CPU使用率峰值](#)

[K2FibAdjMan主機移動進程中的CPU使用率較高](#)

[RkiosPortMan埠稽核進程中的CPU使用率較高](#)

[使用中繼埠連線到IP電話時的CPU使用率較高](#)

[使用RSPAN和第3層控制封包的CPU使用率高](#)

## [用於分析發往CPU的流量的故障排除工具](#)

[工具1：使用SPAN監控CPU流量- Cisco IOS軟體版本12.1\(19\)EW及更高版本](#)

[工具2：內建CPU嗅探器- Cisco IOS軟體版本12.2\(20\)EW及更高版本](#)

[工具3：辨識將流量傳送到CPU的介面- Cisco IOS軟體版本12.2\(20\)EW及更高版本](#)

## [摘要](#)

## [相關資訊](#)

---

# 簡介

本檔案將說明CPU封包處理架構，並介紹如何辨識Catalyst 4500交換器上CPU使用率過高的原因。

# 必要條件

## 需求

本文件沒有特定需求。

## 採用元件

本文中的資訊係根據以下軟體和硬體版本：

- Catalyst 4500 系列交換器
- Catalyst 4948 系列交換器

---



注意：本文檔僅適用於基於Cisco IOS®軟體的交換機。

---

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

## 慣例

如需更多文件慣例的相關資訊，請參閱[思科技術提示慣例](#)。

## 背景資訊

Catalyst 4500系列交換器（包括Catalyst 4948交換器）擁有適用於CPU繫結流量的精密封包處理方式。通常認為這些交換機上的CPU利用率過高是一個問題。本文提供有關CPU封包處理架構的詳細資訊，並說明如何辨識這些交換器上CPU使用率過高的原因。本檔案也列出一些導致Catalyst 4500系列上CPU使用率較高的常見網路或組態案例

在檢視CPU資料包處理架構並解決CPU使用率過高問題之前，您必須瞭解基於硬體的轉發交換機和

基於Cisco IOS軟體的路由器使用CPU的不同方式。常見的誤解是CPU使用率高表示裝置上的資源耗盡和崩潰威脅。容量問題是Cisco IOS路由器上CPU使用率較高的症狀之一。但是，對於基於Catalyst 4500等硬體的轉發交換機，容量問題幾乎從來都不是高CPU使用率的症狀。Catalyst 4500可在硬體專用積體電路(ASIC)中轉送封包，並達到最高1.02億個封包/秒(Mpps)的流量轉送速度。

Catalyst 4500 CPU執行以下功能：

- 管理已設定的軟體通訊協定，例如：
  - 生成樹通訊協定(STP)
  - 路由協定
  - 思科探索通訊協定(CDP)
  - 連線埠彙總通訊協定(PAgP)
  - VLAN中繼線通訊協定(VTP)
  - 動態Trunk通訊協定(DTP)
- 為硬體ASIC的配置/動態條目程式設計，例如：
  - 存取控制清單(ACL)
  - CEF條目
- 內部管理各種元件，例如：
  - 乙太網供電(PoE)線卡
  - 電源
  - 風扇托架
- 管理對交換機的訪問，例如：
  - Telnet
  - 主控台
  - 簡易網路管理通訊協定(SNMP)
- 透過軟體路徑轉送封包，例如：
  - 網際網路封包交換(IPX)路由封包，僅在軟體路徑中支援
  - 最大傳輸單位(MTU)分段

根據此清單，CPU接收或處理資料包可能導致CPU使用率較高。某些為進程傳送的資料包對於網路運行可能至關重要。這些必要封包的範例是跨距樹狀目錄拓撲組態的橋接通訊協定資料單元

(BPDU)。但是，其他資料包可以是軟體轉發的資料流量。這些情況需要交換ASIC將資料包傳送到CPU進行處理：

- 複製到CPU但原始資料包在硬體中交換的資料包。

例如，主機MAC地址學習。

- 傳送到CPU進行處理的資料包

範例包括：

- 路由協定更新
- BPDU
- 有意或無意的流量泛洪

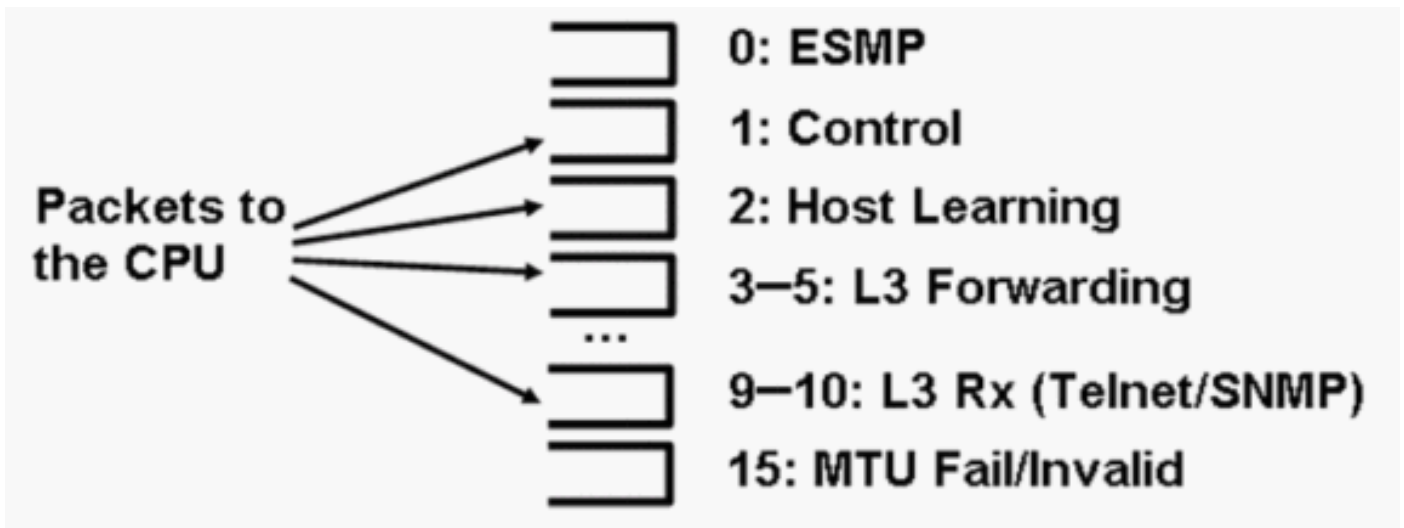
- 傳送到CPU進行轉發的資料包

例如，需要IPX或AppleTalk路由的資料包。

## 瞭解Catalyst 4500 CPU資料包處理架構

Catalyst 4500具有內建的服務品質(QoS)機制，以便區分目的地為CPU的流量型別。該機制基於第2層(L2)/第3層(L3)/第4層(L4)資料包資訊進行區分。Supervisor資料包引擎有16個隊列，用於處理各種型別的資料包或事件。[圖1](#)顯示了這些隊列。[表1](#)列出了這些隊列和每個隊列中排隊的資料包型別。16個隊列允許Catalyst 4500根據資料包型別或優先順序對資料包進行排隊。

圖1 - Catalyst 4500使用多個CPU隊列



Catalyst 4500使用多個CPU隊列

表1 - Catalyst 4500隊列說明

佇列編號	佇列名稱	排隊的資料包
0	EsmP	用於板卡ASIC或其他元件管理的ESMP <sup>1</sup> 資料包 ( 內

		部管理資料包 )
1	控制	L2控制平面資料包，如STP、CDP、PAgP、LACP <sup>2</sup> 或UDLD <sup>3</sup>
2	主機學習	具有未知源MAC地址的幀被複製到CPU以構建L2轉發表
3、4、5	L3 Fwd Highest、L3 Fwd High/Medium、L3 Fwd Low	必須在軟體中轉發的資料包，如GRE <sup>4</sup> 隧道如果目標IP地址的ARP <sup>5</sup> 未解析，資料包將傳送到此隊列。
6、7、8	L2 Fwd Highest、L2 Fwd High/Medium、L2 Fwd Low	橋接後轉發的資料包 <ul style="list-style-type: none"> <li>• 硬體中不支援的協定 ( 例如IPX和AppleTalk路由資料包 ) 會橋接到CPU</li> <li>• ARP請求和響應</li> <li>• 如果因以下原因而無法在硬體中路由資料包，則會橋接具有交換機SVI<sup>6</sup>/L3介面的目標MAC地址的資料包： <ul style="list-style-type: none"> <li>◦ IP報頭選項</li> <li>◦ 過期的TTL<sup>7</sup></li> <li>◦ 非ARPA封裝</li> </ul> </li> </ul>
9、10	L3 Rx High、L3 Rx Low	L3控制平面流量，例如發往CPU IP地址的路由協定。示例包括Telnet、SNMP和SSH <sup>8</sup> 。
11	RPF故障	組播資料包未通過RPF <sup>9</sup> 檢查
12	ACL fwd(snooping)	由DHCP <sup>10</sup> 監聽、動態ARP檢測或IGMP <sup>11</sup> 監聽功能處理的資料包
13	ACL日誌，無法訪問	使用thelogkeyword到達ACE <sup>12</sup> 的資料包或由於輸出ACL中的deny或缺少到目標的路由而被丟棄的資料包這些資料包需要生成ICMP不可達消息。
14	ACL sw處理	由於缺乏安全ACL的其他ACL硬體資源(例如TCAM <sup>13</sup> )而被傳送到CPU的資料包
15	MTU失敗/無效	由於輸出介面MTU大小小於資料包大小而需要分段的資料包

<sup>1</sup>ESMP =甚至簡單管理協定。

<sup>2</sup>LACP =鏈路聚合控制協定。

<sup>3</sup>UDLD =單向鏈路檢測。

<sup>4</sup>GRE =通用路由封裝。

<sup>5</sup>ARP =地址解析協定。

<sup>6</sup>SVI =交換虛擬介面。

<sup>7</sup>TTL =生存時間。

<sup>8</sup>SSH =安全殼協定。

<sup>9</sup>RPF =反向路徑轉發

<sup>10</sup>DHCP =動態主機配置協定。

<sup>11</sup>IGMP = Internet組管理協定。

<sup>12</sup>ACE =訪問控制條目。

<sup>13</sup>TCAM =三重內容可定址儲存器。

這些佇列是不同的佇列：

- L2 Fwd HighestorL3 Fwd Highest
- L2 Fwd高/MediumrL3 Fwd高/中
- L2 Fwd LoworL3 Fwd Low
- L3 Rx HighorL3 Rx Low

資料包根據QoS標籤排隊進入這些隊列，QoS標籤是來自IP服務型別(ToS)的差分服務代碼點(DSCP)值。例如，DSCP為63的資料包會排入L3 Fwd Higheestqueue中。可以在show platform cpu packet statistics allcommand的輸出中看到這16個隊列所收到和丟棄的資料包。此命令的輸出非常長。發出show platform cpu packet statistics命令可以只顯示非零事件。替代命令是show platform cpuport命令。只有在運行Cisco IOS軟體版本12.1(11)EW或更早版本時才使用show platform cpuport命令。此命令已棄用。但是，這一較舊命令在早於Cisco IOS軟體版本12.2(20)EWA的版本中是show tech-supportcommand的一部分。

請將show platform cpu packet statistics命令用於所有故障排除。

<#root>

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, unreachable	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
-----	-----	-----	-----	-----	-----
Esmpl	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, unreachable	0	0	0	0	0
ACL sw processing	0	0	0	0	0
MTU Fail/Invalid	0	0	0	0	0

Catalyst 4500 CPU為表1列出的各種隊列分配權重。CPU根據重要性或型別以及流量優先順序或DSCP來分配權重。CPU根據佇列的相對權重為佇列提供服務。例如，如果控制資料包（例如BPDU）和ICMP回應請求都處於掛起狀態，則CPU會首先為控制資料包提供服務。過多低優先順序或不太重要的流量不會使CPU失去處理或管理系統所需的能力。此機制可確保網路即使在CPU使用率較高的情況下也保持穩定。網路保持穩定的這一能力是您必須瞭解的重要資訊。

Catalyst 4500 CPU資料包處理還有另一個非常重要的實施細節。如果CPU已經服務於高優先順序資料包或進程，但在特定時間段中有更多的備用CPU週期，則CPU服務於低優先順序隊列資料包或執行較低優先順序的後台進程。由於低優先順序資料包處理或後台進程導致的CPU使用率較高，因為CPU會持續嘗試使用所有可用時間，因此被視為正常現象。透過這種方式，CPU將努力在不影響交換機穩定性的情況下實現交換機和網路的最大效能。Catalyst 4500會認為CPU使用率過低，除非CPU在單個時隙中的使用率為100%。

Cisco IOS軟體版本12.2(25)EWA2和更新版本已增強CPU封包和程式處理機制以及計量。因此，請在Catalyst 4500部署上使用這些版本。

## 確定Catalyst 4500上CPU使用率高的原因

現在您已經瞭解Catalyst 4500 CPU資料包處理架構和設計，仍然可以確定為什麼Catalyst 4500 CPU利用率很高。Catalyst 4500具有確定CPU使用率較高的根本原因所必需的命令和工具。確定原因後，管理員可以執行以下任一操作：

- 更正操作-這可能包括更改配置或網路，也可能包括建立用於進一步分析的[Cisco技術支援服務](#)



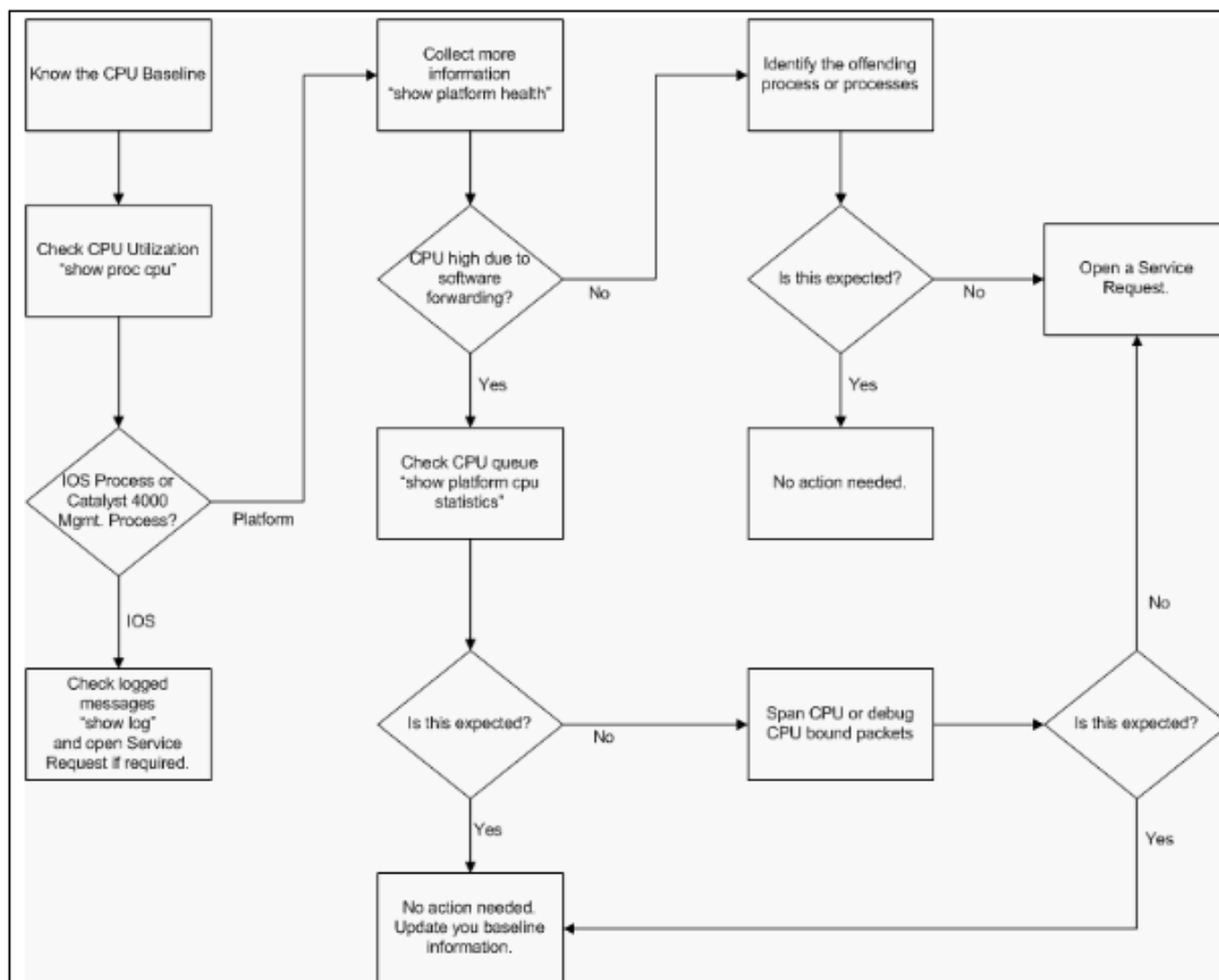
請求。

- 無動作- Catalyst 4500根據預期執行。CPU顯示高CPU利用率，因為Supervisor引擎會將CPU週期最大化，以便執行所有必要的軟體資料包轉發和後台作業。

請務必確定高CPU使用率的原因，即使並非在所有情況下都需要採取糾正措施。高CPU利用率可能只是網路問題的症狀。為了降低CPU使用率，可能需要解決此問題的根本原因。

圖2顯示了用來確定Catalyst 4500 CPU使用率較高的根本原因的故障排除方法。

圖2 - Catalyst 4500交換機上的高CPU利用率故障排除方法



Catalyst 4500交換機上的高CPU利用率故障排除方法

一般故障排除步驟如下：

1. 發出 show processes cpu命令以確定佔用CPU週期的Cisco IOS進程。
2. 發出show platform healthcommand以進一步確定平台特定的進程。
3. 如果高度活躍的進程是K2CpuMan Review，請發出show platform cpu packet statistics命令以確定

傳送到CPU的資料流型別。

如果活動不是由於`x2CpuMan Reviewprocess`引起的，請跳過步驟4，繼續執行步驟5。

4. 透過使用[分析發往CPU的資料流的故障排除工具](#)來確定傳送到CPU的資料包。

故障排除工具的一個示例是CPU交換埠分析器(SPAN)。

5. 請檢視本文檔和[解決常見的高CPU使用率問題](#)部分以瞭解常見原因。

如果仍然無法確定根本原因，請[聯絡Cisco技術支援](#)。

## 稽核CPU使用情況

重要的第一步是瞭解您的配置和網路設定中交換機的CPU使用率。使用`show processes cpucommand`以確定Catalyst 4500交換機上的CPU使用率。隨著您向網路設定中增加更多配置，或者隨著網路流量模式的變化，可能需要持續更新基線CPU利用率。[圖2](#)指明了此要求。

此輸出來自滿載Catalyst 4507R。穩態CPU的利用率約為32%至38%，這是執行此交換機管理功能所必需的：

```
<#root>
```

```
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
<i>!--- Output suppressed.</i>								
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	Cisco IOS ACL Helper
34	0	2	0	0.00%	0.00%	0.00%	0	NAM Manager

五秒CPU使用率表示為：

x%/y%

Thex%代表總CPU使用率，andy%代表在中斷級別消耗的CPU。當您排除Catalyst 4500交換機故障時，僅關注總CPU使用率。

## 瞭解Catalyst 4500交換機上的show processes cpu命令

此show processes cpu輸出顯示有以下兩個進程使用CPU：Cat4k Mgmt HiPriandCat4k Mgmt LoPri。這兩個進程聚合了多個平台特定的進程，這些進程在Catalyst 4500上執行基本的管理功能。這些進程進程控制平面以及需要軟體交換或處理的資料包。

要檢視哪些平台特定的進程在Cat4k Mgmt HiPriandCat4k Mgmt LoPri上下文中使用CPU，請發出show platform healthcommand。

每個平台特定的進程都有CPU的目標/預期使用率。當該進程位於目標內時，CPU在高優先順序環境中執行該進程。show processes cpucommand output counts the utilization underCat4k Mgmt HiPri。如果某個進程超過目標/預期利用率，則該進程將在低優先順序上下文中運行。show processes cpucommand output計算了在執行Cat4k Mgmt LoPri時的該附加使用率。ThisCat4k Mgmt LoPriis還用於運行後台進程和優先順序較低的其他進程，例如一致性檢查和讀介面計數器。此機制允許CPU在必要時運行高優先順序進程，而剩餘的空閒CPU週期用於低優先順序進程。稍微超出目標CPU使用率或短暫的利用率高峰，並不表示存在需要調查的問題。

```
<#root>
```

```
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-JobEventSchedule 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

Ac1-Flattener e 1.00

0.00

10 0 100 500 0

0 0

0:00

KxAc1PathMan create/ 1.00

0.00

10 5 100 500 0

0 0

0:39

```

KxAc1PathMan update      2.00
0.00
   10      0 100 500  0
0  0
  0:00
KxAc1PathMan 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
K2Acce1PacketMan: Tx    10.00
2.20
  20      0 100 500  2
2  1
  82:06
K2Acce1PacketMan: Au     0.10
0.00
   0      0 100 500  0
0  0
  0:00
K2Ac1Man-taggedFlatA     1.00
0.00
  10      0 100 500  0
0  0
  0:00
K2Ac1CamMan stale en     1.00
0.00
  10      0 100 500  0
0  0

```

0:00  
K2Ac1CamMan hw stats 3.00  
1.04  
10 5 100 500 1  
1 0  
39:36  
K2Ac1CamMan kx stats 1.00  
0.00  
10 5 100 500 0  
0 0  
13:40  
K2Ac1CamMan Audit re 1.00  
0.00  
10 5 100 500 0  
0 0  
13:10  
K2Ac1PolicerTableMan 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  
 K2QosDb1Man 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  
RkiosAc1Man 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

```

## 瞭解Catalyst 4500交換機上的show platform health命令

show platform healthcommand提供了許多僅對開發工程師十分有意義的資訊。要對高CPU使用率進行故障排除，請在輸出的%CPU actualcolumn中查詢一個較高的值。此外，請務必檢視該行的右側，以驗證1分鐘和1小時平均值%CPU列中該進程的CPU使用情況。有時，進程會暫時達到峰值，但不會長時間儲存CPU。在硬體程式設計或最佳化程式設計期間，會遇到一些臨時性高CPU利用率。例如，在TCAM中對大型ACL進行硬體程式設計期間，CPU利用率激增是正常的。

在瞭解Catalyst 4500交換機上的show processes cpu命令部分的show platform healthcommand輸出中，Stub-JobEventScheduland K2CpuMan Reviewprocesses使用的CPU週期的數量較高。表2提供了有關顯示在show platform healthcommand的輸出中常見的平台特定進程的一些基本資訊。

表2 -透過show platform health命令顯示的平台特定進程說明

平台特定的程式名稱	說明
Pim審閱	機箱/線卡狀態管理
Ebm	乙太網網橋模組，例如老化和監控
Ac1平面化器/K2Ac1Man	ACL合併過程
KxAclPathMan - PathTagMan-Review	ACL狀態管理和維護
K2CpuMan評論	執行軟體資料包轉發的進程如果看到由於此進程導致CPU使用率較高，請使用show platform cpu packet statistics命令檢視傳送到CPU的資料包。
K2AccelPacketMan	與資料包引擎互動以傳送發往CPU的資料包的驅動程式
K2Ac1CamMan	管理QoS和安全功能的輸入和輸出TCAM硬體
K2Ac1PolicerTableMan	管理輸入和輸出監察器
K2L2	代表Catalyst 4500 Cisco IOS軟體的L2轉發子系統這些進程負責維護各種L2表。
K2PortMan評論	管理各種與埠相關的程式設計功能
K2Fib	FIB <sup>1</sup> 管理
K2FibFlowCache	PBR <sup>2</sup> 快取管理
K2FibAdjMan	FIB鄰接表管理
K2Fib組播	管理多點傳送FIB專案
K2MetStatsMan評論	管理MET <sup>3</sup> 統計
K2QosDb1Man審閱	管理QoS DBL <sup>4</sup>
IrmFibThrottler Thro	IP路由模組
K2 L2老化表Re	管理L2老化函式
GalChassisVp-review	機箱狀態監控
S2w-JobEventSchedule	管理S2W <sup>5</sup> 協定以監控板卡狀態
存根-工作事件排程	基於末節ASIC的線卡監控和維護
RkiosPortMan埠Re	埠狀態監控和維護
Rkios模組狀態R	線路卡監控與維護
EthHoleLinecardMan	在每個板卡中管理GBIC <sup>6</sup>

<sup>1</sup>FIB =轉發資訊庫。

<sup>2</sup>PBR =基於策略的路由。

<sup>3</sup> MET =組播擴展表。

<sup>4</sup>DBL =動態緩衝區限制。

<sup>5</sup>S2W =串列到線。

<sup>6</sup>GBIC =千兆介面轉換器。

## 排除常見的CPU使用率較高問題

本節介紹Catalyst 4500交換機上常見的一些高CPU使用率問題。

### 進程交換資料包造成的CPU使用率高

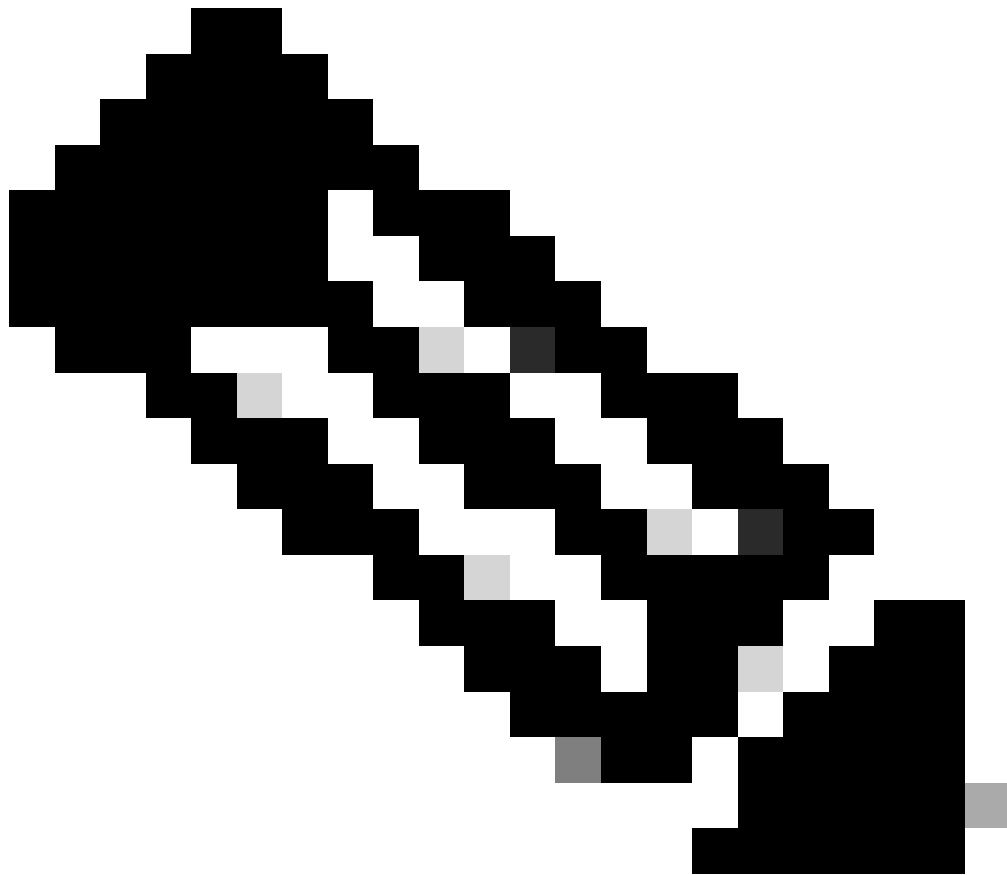
CPU使用率高的常見原因之一是Catalyst 4500 CPU忙於處理用於軟體轉發資料包或控制資料包的資料包。軟體轉發資料包的示例包括IPX或控制資料包，例如BPDU。這些資料包中通常只有一小部分會傳送到CPU。但是，始終有大量資料包可能表示配置錯誤或網路事件。您必須確定導致將資料包轉發到CPU進行處理的事件原因。此辨識可讓您偵錯高CPU使用率問題。

由於進程交換資料包而導致CPU使用率較高的常見原因包括：

- [大量的生成樹埠例項](#)
- [ICMP重定向；在同一介面上路由資料包](#)
- [IPX或AppleTalk路由](#)
- [主機學習](#)
- [安全ACL的硬體資源不足\(TCAM\)](#)
- [ACL中的Theologkeyword](#)
- [第2層轉發環路](#)

將資料包交換到CPU的其他原因包括：

- MTU分段-請確保資料包路徑上的所有介面具有相同的MTU。
- ACL帶有established以外的TCP標誌
- IP第6版(IPv6)路由-僅透過軟體交換路徑支援此功能。
- GRE -僅透過軟體交換路徑支援此功能。
- 拒絕輸入或輸出路由器ACL中的流量(RACL)



注意：在Cisco IOS軟體版本12.1(13)EW1及更高版本中，速率受到限制。

---

在ACL的介面下發出o ip unreachablecommand。

- 由於大量直接連線的主機，過多的ARP和DHCP流量進入CPU進行處理

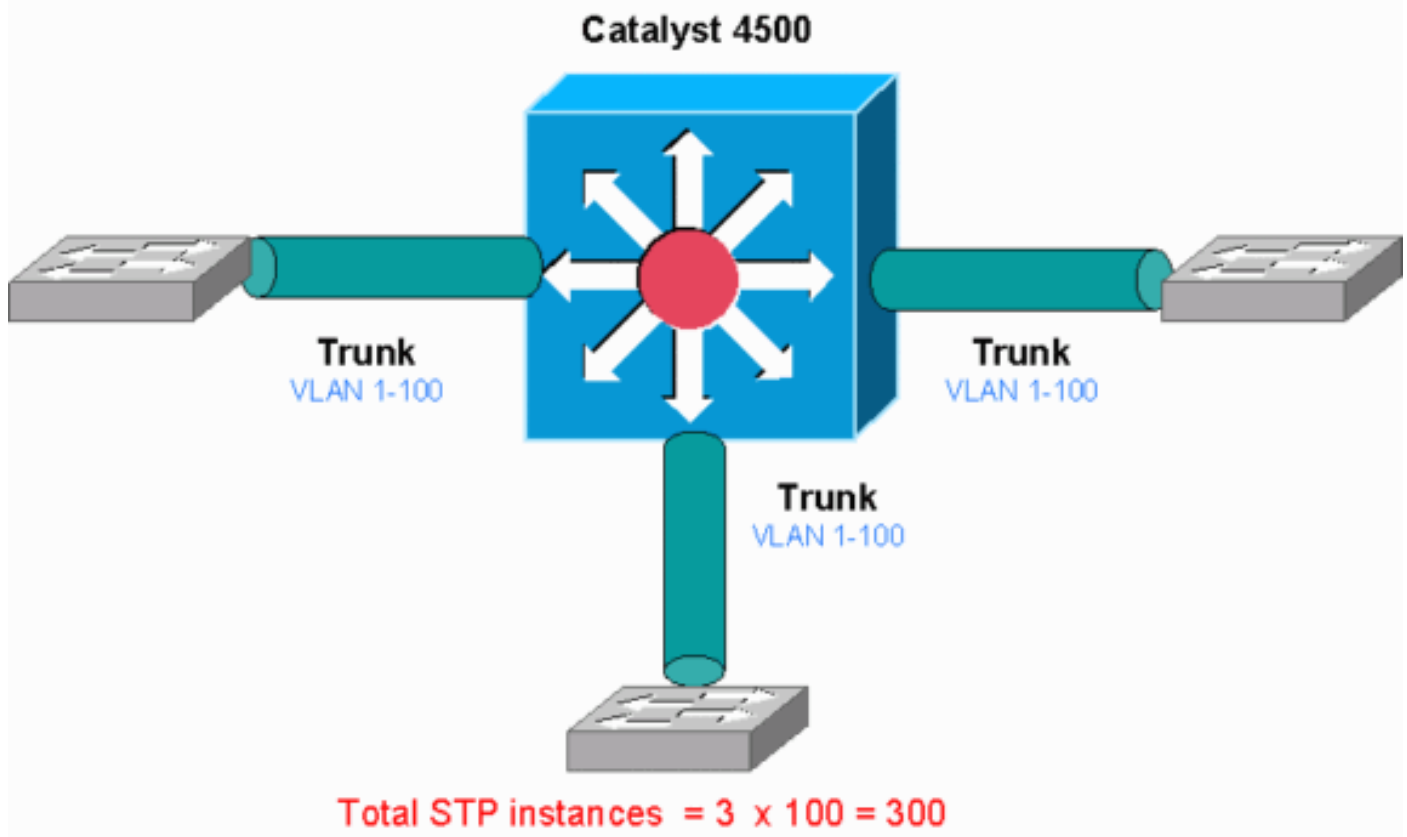
如果懷疑DHCP攻擊，請使用DCHP監聽來限制來自任何特定主機埠的DHCP流量。

- 合法或行為不當的終端站進行過多的SNMP輪詢

#### 大量的生成樹埠例項

Catalyst 4500在每VLAN生成樹+ (PVST+)模式下支援3000個生成樹埠例項或活動埠。除Supervisor引擎II+和II+TS以及Catalyst 4948外，所有管理引擎都支援該功能。Supervisor引擎II+和II+TS以及Catalyst 4948支援多達1500個埠例項。如果超出這些STP例項建議，交換機的CPU使用率就會很高。





Catalyst 4500

下圖顯示一台Catalyst 4500帶有三個中繼埠，每個埠承載VLAN 1到100。這相當於300個生成樹埠例項。一般而言，您可以使用此公式計算生成樹埠例項：

Total number of STP instances = Number of access ports + Sum of all VLANs that are carried in each of the trunks

在圖中，沒有接入埠，但三個中繼承載VLAN 1至100：

Total number of STP instances = 0 + 100 + 100 + 100 = 300

**第1步：** 使用 `show processes cpu` 命令檢查Cisco IOS進程。

本節介紹管理員使用的命令，以縮小CPU使用率過高的問題。如果發出`show processes cpu`命令，則可以看到兩個主進程(Cat4k Mgmt LoPriandSpanning Tree)對CPU的使用率較高。僅憑此資訊，您便知道生成樹進程佔用了大部分CPU週期。

<#root>

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
<i>!--- Output suppressed.</i>								
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
<i>!--- Output suppressed.</i>								
27	158796	59219	2681	32.55%	33.80%	21.43%	0	Cat4k Mgmt LoPri
<i>!--- Output suppressed.</i>								
28	20	1693	11	0.00%	0.00%	0.00%	0	Galios Reschedul
29	0	1	0	0.00%	0.00%	0.00%	0	Cisco 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
<i>!--- Output suppressed.</i>								
43	78564	20723	3791	32.63%	30.03%	17.35%	0	Spanning Tree
<i>!--- Output suppressed.</i>								
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

第2步：使用show platform health命令檢查Catalyst 4500特定的進程。

要瞭解哪個平台特定的進程佔用了CPU，請發出show platform healthcommand。從該輸出中，可以看到k2CpuMan Reviewprocess (處理計算密集型資料包的作業)用盡了CPU：

<#root>

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

K2Acce1PacketMan: Tx 10.00  4.95    20     0  100  500    5  4  0  0:36
K2Acce1PacketMan: Au  0.10  0.00     0     0  100  500    0  0  0  0:00
K2Ac1Man-taggedFlatA  1.00  0.00    10     0  100  500    0  0  0  0:00
```

第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。

發出show platform cpu packet statistics命令以檢查哪一個CPU隊列收到了計算密集型資料包。本部分中的輸出顯示控制隊列收到了很多資料包。請使用[表1](#)中的資訊和[步驟1](#)中做出的結論。您可以確定CPU處理的資料包和CPU使用率高的原因是BPDU處理。

<#root>

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
-----	-----	-----	-----	-----	-----
Esmpl	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

第4步：確定根本原因。

發出show spanning-tree summary命令。您可以檢查收到BPDU是否是因為生成樹埠例項的數量過多。輸出清楚地指出了根本原因：

<#root>

Switch#

show spanning-tree summary

```
Switch is in pvst mode
Root bridge for: none
Extended system ID      is enabled
Portfast Default        is disabled
PortFast BPDUGuard Default is disabled
Portfast BPDUGuard 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

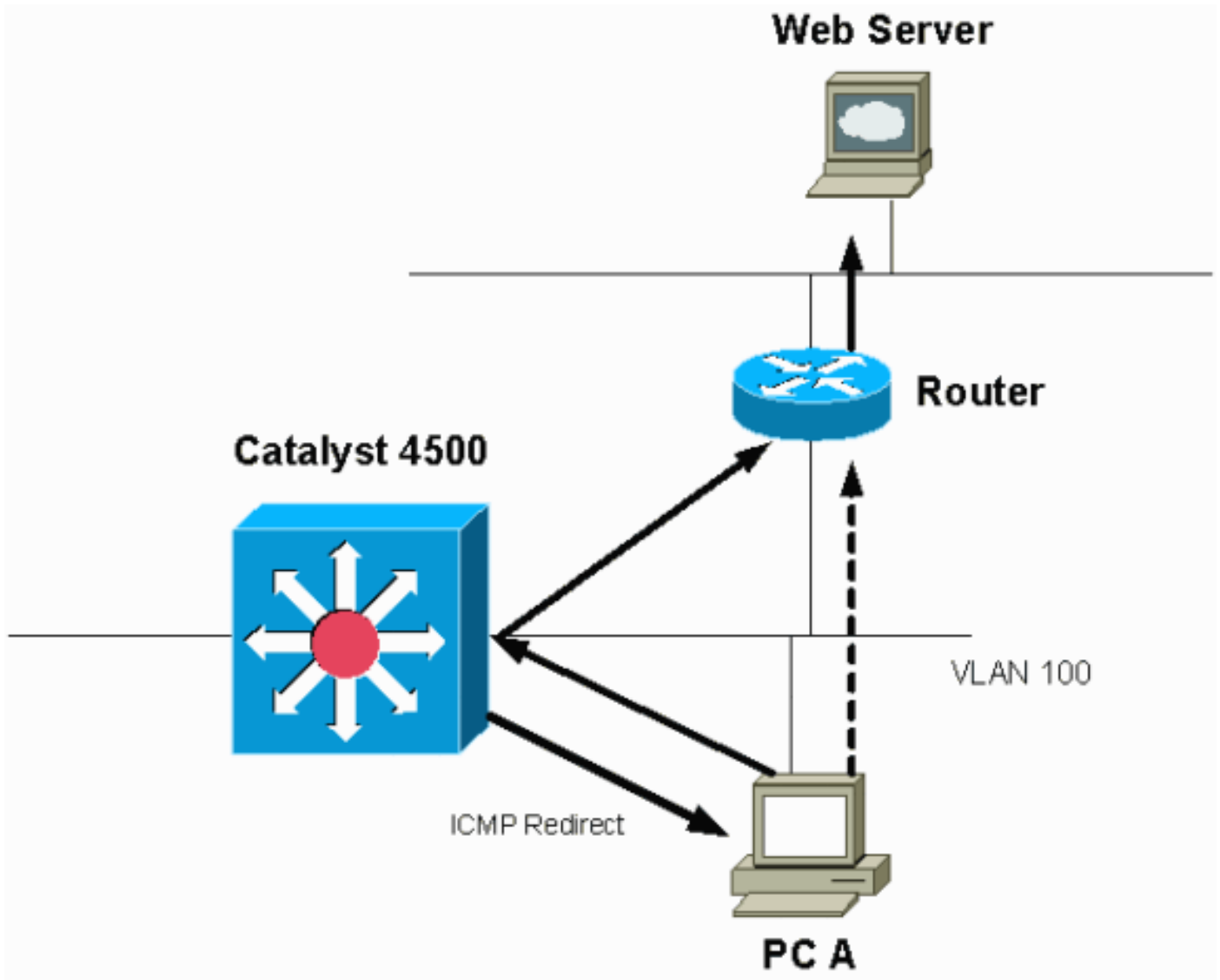
有大量VLAN採用PVST+模式配置。為了解決此問題，請將STP模式更改為多生成樹(MST)。在某些情況下，由於在所有中繼埠上轉發大量VLAN，STP例項的數量會很高。在這種情況下，請手動修剪中繼中不需要的VLAN，以使STP活動埠數降到遠低於建議值的水準。

**提示：**請確保未將IP電話埠配置為中繼埠。這是一種常見的配置錯誤。使用語音VLAN配置配置IP電話埠。此配置會建立偽中繼，但不需要您手動修剪不必要的VLAN。有關如何配置語音埠的詳細資訊，請參閱[配置語音](#)介面軟體配置指南。非Cisco IP電話不支援此語音VLAN或輔助VLAN配置。必須使用非Cisco IP電話手動修剪埠。

ICMP重定向；在同一介面上路由資料包

在同一介面上路由封包，或是在同一L3介面上輸出和輸入流量，都可能導致交換器進行ICMP重新導向。如果交換器知道通往最終目的地的下一躍點裝置位於與傳送裝置相同的子網路中，便會產生ICMP重新導向至來源。重定向消息指示源將資料包直接傳送到下一跳裝置。這些消息表明下一跳裝置具有到達目的地的更好路由，該路由比此交換機少一跳。

在本節的圖表中，PC A與Web伺服器通訊。PC A的預設網關指向VLAN 100介面IP地址。但是，使Catalyst 4500到達目的地的下一跳路由器與PC A位於同一子網中。在這種情況下，最佳路徑是直接傳送到「Router」。Catalyst 4500向PC A傳送ICMP重定向消息。該消息指示PC A透過路由器（而不是Catalyst 4500）將資料包傳送到Web伺服器。但是，大多數情況下，終端裝置不會響應ICMP重定向。缺乏回應導致Catalyst 4500花費大量CPU週期為所有Catalyst透過與輸入封包相同的介面轉送的封包產生這些ICMP重新導向。



預設情況下，*ICMP*重定向已啟用。

預設情況下，*ICMP*重定向已啟用。要停用它，請使用`ip icmp redirects`命令。在相關SVI或L3介面下發出命令。



注意：由於ip icmp redirects 是預設命令，因此它在show running-configuration命令輸出中是不可見的。

---

第1步：使用 show processes cpu 命令檢查Cisco IOS進程。

發出show processes cpu命令。您可以看到兩個主進程(Cat4k Mgmt LoPriandIP Input)對CPU的使用率較高。僅憑此資訊，您就知道IP資料包的處理會佔用相當大一部分的CPU。

<#root>

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. show platform health 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
----	-----------	-----------	-----	--------	--------	--------	---	----------

第2步：使用 show platform health 命令檢查Catalyst 4500特定的進程。



show platform health命令的輸出可以證實，使用CPU是為了處理計算密集型資料包。

```
<#root>
```

```
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
K2Acce1PacketMan: Tx  10.00   4.95    20     0  100  500     5  4    0  0:36
K2Acce1PacketMan: Au   0.10   0.00     0     0  100  500     0  0    0  0:00
K2Ac1Man-taggedFlatA   1.00   0.00    10     0  100  500     0  0    0  0:00
```

第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。

發出show platform cpu packet statistics命令以檢查哪一個CPU隊列收到了計算密集型資料包。可以看到L3 Fwd Lowqueue收到很多資料流。

```
<#root>
```

```
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
<b>L3 Fwd Low</b>	<b>4717094264</b>	<b>3841</b>	<b>3879</b>	<b>3873</b>	<b>3547</b>
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

第4步：確定根本原因。

在這種情況下，請使用CPU SPAN來確定傳送到CPU的資料流。有關CPU SPAN的資訊，請參閱本文檔的[工具1：使用SPAN監控CPU資料流- Cisco IOS軟體版本12.1\(19\)EW和更高版本](#)部分。使用show running-configuration命令完成資料流分析和配置。在這種情況下，封包會透過相同介面路由，導致每個封包都發出ICMP重新導向。此根本原因是Catalyst 4500上CPU使用率較高的常見原因之一。

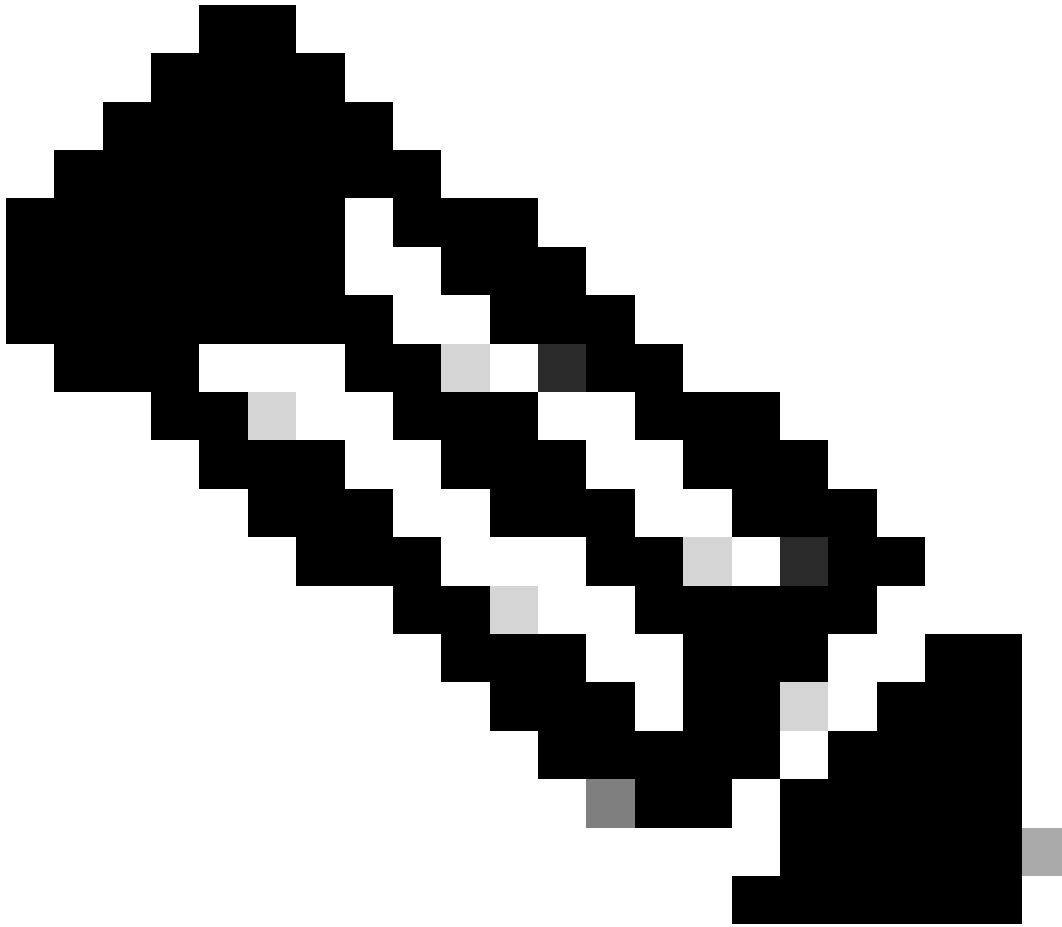
您可以預期sourcing裝置會對Catalyst 4500傳送的ICMP重新導向做出反應，並變更目的地的下一躍點。但是，並非所有裝置都會響應ICMP重定向。如果裝置未響應，Catalyst 4500必須為交換機從傳送裝置接收的每個資料包傳送重定向。這些重定向會消耗大量CPU資源。解決方案是停用ICMP重定向。在介面下，發出no ip redirects命令。

當還配置了輔助IP地址時，可能會出現這種情況。當您啟用次要IP位址時，IP重新導向會自動停用。請確保您未手動啟用IP重定向。

如[此ICMP重定向：在同一介面上路由資料包](#)所示，大多數終端裝置不會對ICMP重定向消息做出響應。因此，一般情況下停用此功能。

#### IPX或AppleTalk路由

Catalyst 4500僅透過軟體轉發路徑支援IPX和AppleTalk路由。設定此類通訊協定時，較高的CPU使用率是正常的。



注意：交換同一VLAN中的IPX和AppleTalk資料流不需要進程交換。只有需要路由的資料包需要軟體路徑轉發。

---

第1步：使用 `show processes cpu` 命令檢查Cisco IOS進程。

發出`show processes cpu` 命令以檢查哪一個Cisco IOS進程佔用了CPU。在此命令輸出中，請注意CPU使用率最高的進程是`theCat4k Mgmt LoPri`：

<#root>

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
----	-----------	-----------	-----	--------	--------	--------	---	------------------

第2步：使用 show platform health 命令檢查Catalyst 4500特定的進程。

show platform health命令的輸出可以證實，使用CPU是為了處理計算密集型資料包。

<#root>

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

```

第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。

要確定傳送到CPU的資料流型別，請發出show platform cpu packet statistics命令。

```
<#root>
```

```
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
<b>L2 Fwd Low</b>	<b>576905398</b>	<b>1837</b>	<b>1697</b>	<b>1938</b>	<b>1515</b>

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, unreachable	11013420	9	8	8	8

第4步：確定根本原因。

由於管理員配置了IPX或AppleTalk路由，因此根本原因的確定必須直截了當。但是為了確認，請跨越CPU流量並確保您看到的流量是預期的流量。有關CPU SPAN的資訊，請參閱本文檔的[工具1：使用SPAN監控CPU資料流- Cisco IOS軟體版本12.1\(19\)EW和更高版本](#)部分。

在這種情況下，管理員必須將基準CPU更新為當前值。當CPU處理軟體交換資料包時，Catalyst 4500 CPU按預期工作。

### 主機學習

如果MAC地址表中沒有MAC地址，則Catalyst 4500會獲取不同主機的MAC地址。交換引擎將包含新MAC地址的資料包副本轉發到CPU。

所有VLAN介面（第3層）都使用機箱基本硬體地址作為其MAC地址。因此，MAC位址表中沒有專案，且目的地為這些VLAN介面的封包不會傳送到CPU進行處理。

如果交換機需要瞭解的新MAC地址過多，則可能導致CPU使用率較高。

第1步：使用 `show processes cpu` 命令檢查Cisco IOS進程。

發出 `show processes cpu` 命令以檢查哪一個Cisco IOS進程佔用了CPU。在此命令輸出中，請注意CPU使用率最高的進程是 `theCat4k Mgmt LoPri`：

```
<#root>
```

```
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

```

第2步：使用show platform health命令檢查Catalyst 4500特定的進程。

show platform healthcommand的輸出可以證實，使用CPU是為了處理計算密集型資料包。

```
<#root>
```

```
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
```

第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。

要確定傳送到CPU的資料流型別，請發出show platform cpu packet statistics命令。

```
<#root>
```

```
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, unreachable	11013420	9	8	8	8

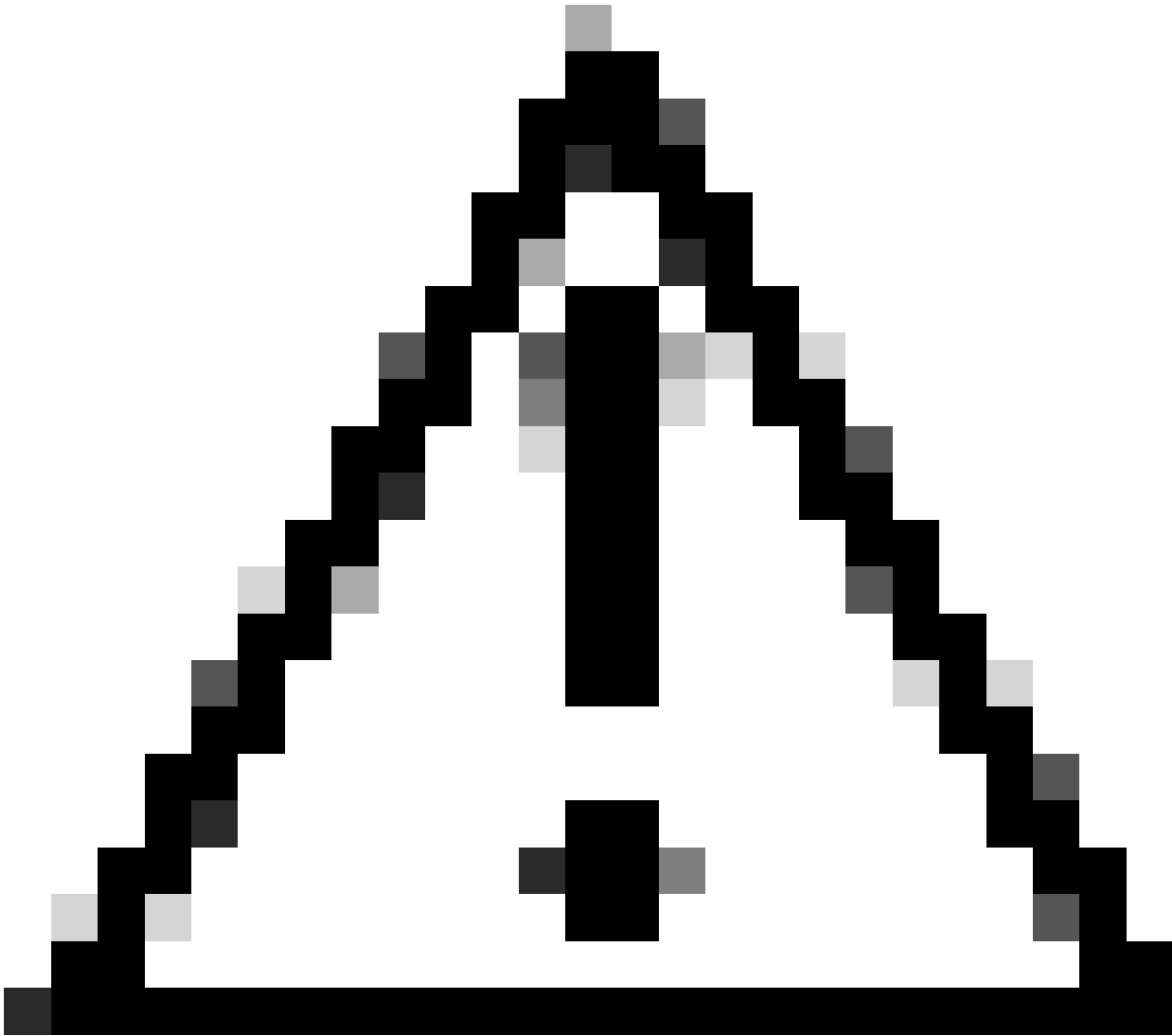
第4步：確定根本原因。

show platform health命令的輸出顯示CPU發現很多新MAC地址。這種情況通常是由網路拓撲不穩定引起的。例如，如果生成樹拓撲發生變化，交換機將生成拓撲更改通知(TCN)。在PVST+模式下，TCN問題可將老化時間縮短到15秒。如果在該時間段內未重新獲知地



址，則會刷新MAC地址條目。在快速STP (RSTP) (IEEE 802.1w)或MST (IEEE 802.1s)的情況下，如果TCN來自另一台交換機，則條目會立即老化。這種老化會導致重新獲取MAC地址。如果拓撲很少變化，則這不是主要問題。但是，由於鏈路抖動、交換機故障或未啟用PortFast的主機埠，可能會出現過多的拓撲更改。可能會進行大量MAC表刷新和後續重新學習。確定根本原因的下一個步驟是排除網路故障。交換器會依預期運作，並將封包傳送到CPU以進行主機位址學習。辨識並修復導致過多TCN的故障裝置。

您的網路可能有許多裝置以突發方式傳送流量，這會導致MAC地址老化，隨後在交換機上重新獲取。在這種情況下，請增加MAC地址表老化時間，以便提供一些緩衝。隨著老化時間的延長，交換機在老化之前將裝置MAC地址在表中保留更長時間。



**注意：**僅在仔細考慮之後才進行此過時更改。如果您的網路中有流動裝置，此更改可能導致流量黑洞。

Catalyst 4500使用Cisco TCAM對已配置的ACL進行程式設計。TCAM允許在硬體轉發路徑中應用ACL。無論轉發路徑中是否包含ACL，對交換機的效能均沒有影響。無論ACL的大小如何，效能都是恆定的，因為ACL查詢的效能是線速的。但是，TCAM資源有限。因此，如果配置過多的ACL條目，您將超過TCAM容量。[表3](#)顯示了每一台Catalyst 4500 Supervisor引擎和交換機上的可用TCAM資源數。

表3 - Catalyst 4500 Supervisor引擎/交換機上的TCAM容量

產品	特徵TCAM ( 每個方向 )	QoS TCAM ( 每個方向 )
Supervisor引擎II+/II+TS	8192個條目帶有1024個掩碼	8192個條目帶有1024個掩碼
管理引擎III/IV/V和Catalyst 4948	16,384個條目帶有2048個掩碼	16,384個條目帶有2048個掩碼
管理引擎V-10GE和Catalyst 4948-10GE	16,384個條目，帶有16,384個掩碼	16,384個條目，帶有16,384個掩碼

交換機使用功能TCAM對安全ACL進行程式設計，例如RACL和VLAN ACL (VACL)。交換器也使用TCAM功能提供安全功能，例如用於動態ACL的IP來源防護(IPSG)。交換機使用QoS TCAM對分類和監察器ACL進行程式設計。

當在安全ACL程式設計期間Catalyst 4500耗盡TCAM資源時，ACL的部分應用將透過軟體路徑進行。到達這些ACE的資料包在軟體中處理，這會導致CPU使用率較高。ACL是由上而下程式設計。換句話說，如果ACL不適合TCAM，則ACL底部的ACE可能不會在TCAM中進行程式設計。

當TCAM溢位發生時，會出現以下警告消息：

```
%C4K_HWACLMAN-4-ACLHWPROGERRREASON: (Suppressed 1times) Input(null, 12/Normal)
Security: 140 - insufficient hardware TCAM masks.
%C4K_HWACLMAN-4-ACLHWPROGERR: (Suppressed 4 times) Input Security: 140 - hardware TCAM
limit, some packet processing can be software switched.
```

可以在show loggingcommand輸出中看到此錯誤消息。該消息最終表明可能會進行某些軟體處理，從而可能導致CPU使用率較高。



注意：如果更改了大型ACL，在TCAM中再次程式設計更改的ACL之前，您可能會看到此消息。

---

第1步：使用show processes cpu命令檢查Cisco IOS進程。

發出show processes cpucommand。您可以看到CPU使用率較高，因為Cat 4k Mgmt LoPriprocess佔用了大部分CPU週期。

<#root>

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
----	-------	---------	----	-------	-------	-------	---	------------------

第2步：使用 show platform health 命令檢查Catalyst 4500特定的進程。

發出show platform health命令。可以看到**K2CpuMan Review** (處理計算密集型資料包的作業) 使用了CPU。

```
<#root>
```

```
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-JobEventSchedule 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
Ac1-Flattener 1.00 0.00 10 5 100 500 0 0 0 0:04
KxAc1PathMan create/ 1.00 0.00 10 5 100 500 0 0 0 0:21
KxAc1PathMan update 2.00 0.00 10 6 100 500 0 0 0 0:05
KxAc1PathMan 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

```

```

K2Acce1PacketMan: Tx 10.00 2.30 20 0 100 500 2 2 2 1345:30
K2Acce1PacketMan: Au 0.10 0.00 0 0 100 500 0 0 0 0:00

```

第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。

您需要進一步瞭解哪個CPU隊列，以及哪些型別的流量會到達CPU隊列。發出**show platform cpu packet statistics**命令。可以看到ACL SW processingqueue收到大量的資料包。因此，TCAM溢位是CPU利用率過高問題的原因。

```
<#root>
```

```
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

<b>ACL sw processing</b>	<b>842889240</b>	<b>1453</b>	<b>1532</b>	<b>1267</b>	<b>1179</b>
--------------------------	------------------	-------------	-------------	-------------	-------------

#### 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

第4步：解決問題。

在步驟3中，已確定了發生這種情況的根本原因。刪除導致溢位的ACL或將ACL最小化以避免溢位。此外，請檢視[使用ACL配置網路安全配置指南](#)，以最佳化硬體中的ACL配置和程式設計。

#### ACL中的log關鍵字

Catalyst 4500支援記錄命中任何特定ACL條目的資料包詳細資訊，但過多的記錄可能導致高CPU利用率。除在流量發現階段之外，請避免使用oflogkeywords。在流量發現階段，您可以辨識流經網路的流量，您尚未為其明確配置ACE。請勿使用thelogkeyword來收集統計資訊。在Cisco IOS軟體版本12.1(13)EW和更新版本中，像片訊息受到速率限制。如果您使用elogmessages對與ACL匹配的資料包數量進行計數，則該計數是不準確的。請改用show access-list命令以獲得準確的統計資訊。由於檢視配置orlogmessages可指示使用ACL日誌記錄功能，因此更容易辨識此根本原因。

第1步：使用show processes cpu命令檢查Cisco IOS進程。

發出show processes cpu命令以檢查哪一個Cisco IOS進程佔用了CPU。在此命令輸出中，可發現CPU使用率最高的進程是theCat4k Mgmt LoPri：

```
<#root>
```

```
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
----	-------	---------	----	-------	-------	-------	---	------------------

第2步：使用show platform health命令檢查Catalyst 4500特定的進程。

檢查使用CPU的平台特定進程。發出show platform health命令。在輸出中，請注意k2CpuMan Reviewprocess使用了大部分CPU週期。本練習表明CPU在處理髮往它的資料包時處於忙碌狀態。

<#root>

Switch#

```
show platform health
```

	%CPU Target	%CPU Actual	RunTimeMax Target	RunTimeMax Actual	Priority Fg	Priority Bg	Average 5Sec	%CPU Min	%CPU Hour	Total 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-JobEventSchedule	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
Ac1-Flattener        1.00  0.00   10    5  100  500    0  0    0  0:04
KxAc1PathMan create/ 1.00  0.00   10    5  100  500    0  0    0  0:21
KxAc1PathMan update  2.00  0.00   10    6  100  500    0  0    0  0:05
KxAc1PathMan 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

K2Acce1PacketMan: Tx 10.00  2.30    20    0  100  500    2  2    2 1345:30
K2Acce1PacketMan: Au  0.10  0.00    0    0  100  500    0  0    0  0:00

```

第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別。

要確定傳送到CPU的資料流型別，請發出 `show platform cpu packet statistics` 命令。在此命令輸出中，可以看到收到資料包是由於 ACLlogkeyword：

```
<#root>
```

```
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, unreachable	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, unreachable	193094788	509	362	437	394

第4步：解決問題。

在步驟3中，已確定了發生這種情況的根本原因。若要避免此問題，請從ACL中移除thelogkeyword。在Cisco IOS軟體版本12.1(13)EW1和更新版本中，封包是受速率限制的，因此CPU使用率不會變得太高。使用訪問清單計數器作為跟蹤ACL命中的方法。可以在 `show access-list acl_id`命令輸出中看到訪問清單計數器。

#### 第2層轉發環路

生成樹協定(STP)實施不佳以及各種可能會影響STP的問題，都可能導致第2層轉發環路。

第1步：使用`show processes cpu`命令檢查Cisco IOS進程

本節介紹管理員使用的命令，以縮小CPU使用率過高的問題。如果發出`show processes cpu`命令，則可以看到兩個主進程(Cat4k Mgmt LoPriandSpanning Tree)對CPU的使用率較高。僅憑此資訊，您便知道生成樹進程佔用了大部分CPU週期。

```
<#root>
```

```
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

第2步：使用show platform health命令檢查Catalyst 4500特定的進程

要瞭解哪個平台特定的進程佔用了CPU，請發出show platform health命令。從該輸出中，可以看到k2CpuMan Reviewprocess (處理計算密集型資料包的作業) 用盡了CPU：

<#root>

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

K2Acce1PacketMan: Tx 10.00  4.95    20     0  100  500    5  4  0  0:36
K2Acce1PacketMan: Au  0.10  0.00     0     0  100  500    0  0  0  0:00
K2Ac1Man-taggedFlatA  1.00  0.00    10     0  100  500    0  0  0  0:00
```

第3步：檢查接收流量的CPU隊列，以確定計算密集型流量的型別

發出show platform cpu packet statistics命令以檢查哪一個CPU隊列收到了計算密集型資料包。本部分中的輸出顯示控制隊列收到了很多資料包。請使用表1中的資訊和步驟1中做出的結論。您可以確定CPU處理的資料包和CPU使用率高的原因是BPDU處理。

```
<#root>
```

```
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
-----
```

Esmpl	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

#### 第4步：確定根本原因並解決問題

一般而言，您可以完成以下步驟以進行疑難排解（視情況而定，某些步驟並非必要）：

- 辨識回圈。
- 發現環路的範圍。
- 中斷回圈。
- 修正回圈的原因。
- 恢復冗餘。

每一個步驟都在[排除轉發環路故障-排除運行Cisco IOS系統軟體的Catalyst交換機上的STP故障](#)中進行了詳細說明。

## 第5步：實施高級STP功能

- 

**BDPU防護** -保護STP，以防未經授權的網路裝置連線到已啟用portfast的埠。有關詳細資訊，請參閱[生成樹PortFast BPDU防護增強功能](#)。

- 

**環路防護** -提高第2層網路的穩定性。有關詳細資訊，請參閱[使用環路防護和BPDU遲滯檢測功能的生成樹協定增強功能](#)。

- 

**根防護** -強制在網路中放置根網橋。有關詳細資訊，請參閱[生成樹協定根防護增強功能](#)。

- 

**UDLD** -檢測單向鏈路並防止形成轉發環路。有關詳細資訊，請參閱[瞭解和配置單向鏈路檢測協定功能](#)。

## CPU使用率高的其他原因

以下是導致CPU使用率較高的其他已知原因：

- 

[過度連結翻動](#)

- 

[由於FIB一致性檢查導致CPU使用率峰值](#)

- 

[K2FibAdjMan Host Moveprocess中的CPU使用率較高](#)

- 

[RkiosPortMan Port Reviewprocess中的CPU使用率較高](#)

-

[使用中繼埠連線到IP電話時的CPU使用率較高](#)

•

[使用RSPAN和第3層控制封包的CPU使用率高](#)

•

大型ACL程式設計期間的峰值

CPU使用率的激增發生在應用或從介面刪除大型ACL的過程中。

過度連結翻動

當一個或多個連線的鏈路開始過度抖動時，Catalyst 4500顯示較高的CPU利用率。此情況發生在早於Cisco IOS軟體版本12.2(20)EWA的Cisco IOS軟體版本中。

第1步：使用show processes cpu命令檢查Cisco IOS進程。

發出show processes cpu command以檢查哪一個Cisco IOS進程佔用了CPU。在此命令輸出中，請注意CPU使用率最高的進程是theCat4k Mgmt LoPri：

<#root>

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
```

第2步：使用show platform health命令檢查Catalyst 4500特定的進程。

show platform healthcommand的輸出指明KxAc1PathMan createprocess用盡了CPU。此程式是用來建立內部路徑。

```
<#root>
```

```
Switch#
```

```
show platform health
```

	%CPU Target	%CPU Actual	RunTimeMax Target	RunTimeMax Actual	Priority Fg	Priority Bg	Average 5Sec	%CPU Min	%CPU Hour	Total 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-JobEventSchedule	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
Ac1-Flattener	1.00	0.00	10	5	100	500	0	0	0	0:00
<b>KxAc1PathMan create/</b>	<b>1.00</b>	<b>69.11</b>	<b>10</b>	<b>5</b>	<b>100</b>	<b>500</b>	<b>42</b>	<b>53</b>	<b>22</b>	<b>715:0</b>
KxAc1PathMan update	2.00	0.76	10	6	100	500	0	0	0	86:00
KxAc1PathMan 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
K2Acce1PacketMan: Tx	10.00	6.86	20	0	100	500	5	5	4	78:42

第3步：確定根本原因。

啟用連結啟動/關閉訊息的記錄。預設情況下不啟用此日誌記錄。該功能可幫助您快速縮小違規連結。在所有介面下發出**the logging event link-status**command。可以使用**interface range**command方便地對一系列介面啟用，如以下示例所示：

```
<#root>
```

```
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
```



```
<#root>
```

```
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
```

辨識出故障或抖動介面後，請關閉介面以解決CPU使用率較高的問題。Cisco IOS軟體版本12.2(20)EWA和更新版本已針對此抖動連結情況改善Catalyst 4500的行為。因此，對CPU的影響沒有改進前那麼大。請記住，此過程是後台進程。此問題造成的CPU使用率高不會對Catalyst 4500交換機造成負面影響。

FIB一致性檢查導致的CPU使用率峰值

在FIB表一致性檢查期間，Catalyst 4500可能會顯示CPU使用率的瞬時峰值。FIB表是CEF進程建立的L3轉發表。一致性檢查可維護Cisco IOS軟體FIB表和硬體條目之間的一致性。這種一致性可確保資料包不會發生錯誤路由。該檢查每2秒執行一次，並且作為低優先順序的後台進程運行。此進程是正常行為，不會干擾其他高優先順序進程或資料包。

**show platform health** command的輸出顯示**K2Fib Consistency**佔用了大部分的CPU。

**注意：**此處理的平均CPU使用率在一分鐘或一小時內微不足道，這可以確認此檢查是短暫的定期檢查。此後台進程僅使用空閒CPU週期。

```
<#root>
```

```
Switch#
```

```
show platform health
```

	%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				

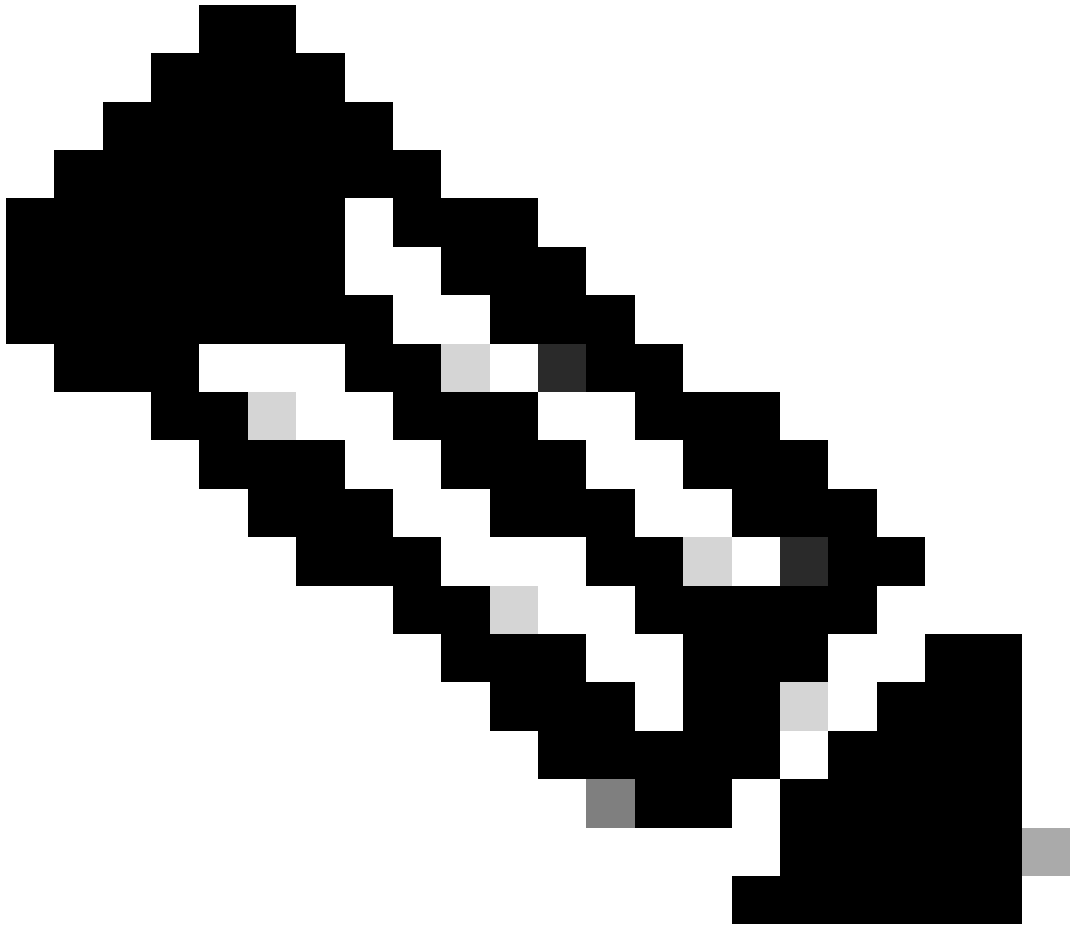
0 0 0

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

K2FibAdjMan主機移動進程中的CPU使用率較高

Catalyst 4500可能會顯示K2FibAdjMan Host Moveprocess中的CPU使用率較高。這一高使用率顯示在show platform healthcommand的輸出中。許多MAC地址經常過期或在新埠上獲知，這會導致此高CPU使用率。mac-address-table aging-time的預設值為5分鐘或300秒。此問題的解決方法是增加MAC地址老化時間，或者您可以設計網路以避免大量的MAC地址移動。Cisco IOS軟體版本12.2(18)EW和更新版本已增強此程式行為，以便減少耗用CPU。請參閱Cisco bug [IDCSCed15021 \( 僅限註冊使用者 \)](#)。



附註：只有完成註冊的思科使用者能存取思科內部工具與資訊。

---

<#root>

Switch#

show platform health

	%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

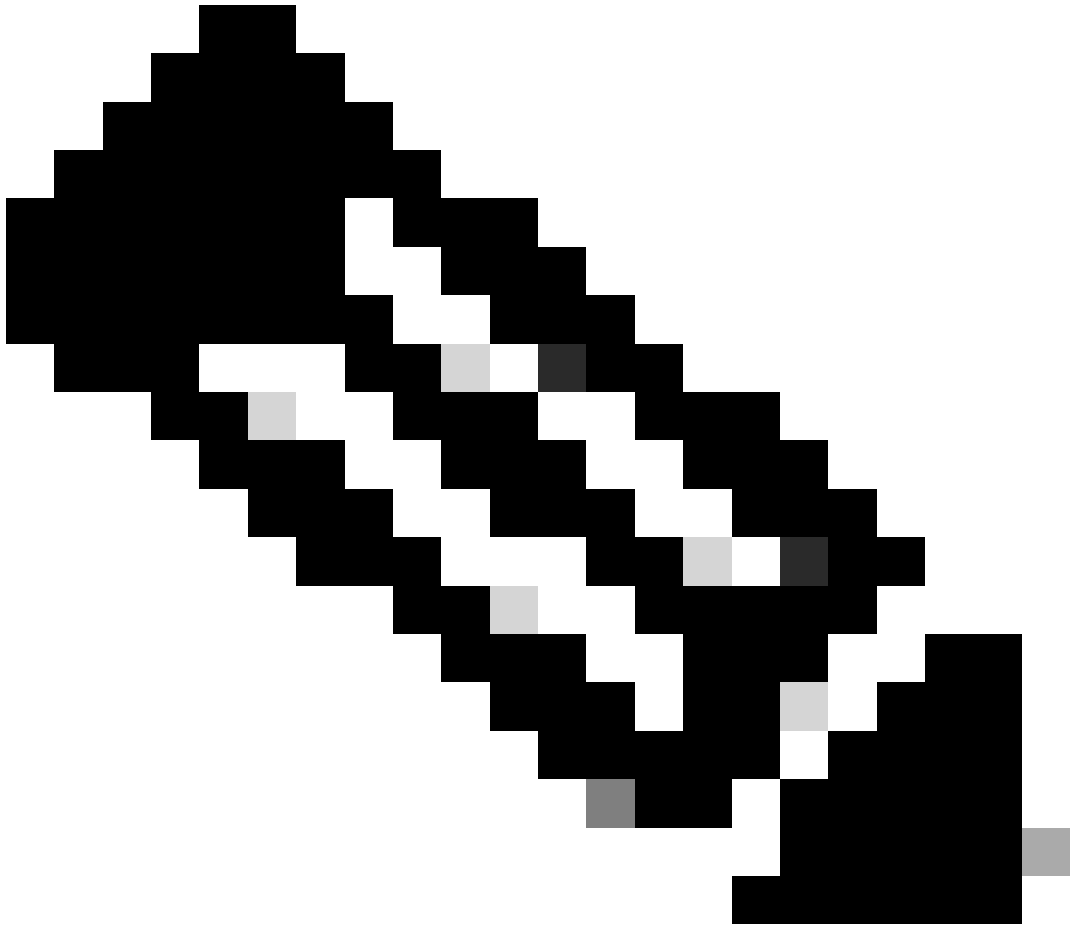
!--- Output suppressed.

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

您可以在全局配置模式下修改MAC地址的最大老化時間。命令語法是 `mac-address-table aging-time 秒` (對於路由器) 和 `mac-address-table aging-time seconds [vlan vlan-id]` (對於Catalyst交換機)。有關詳細資訊，請參閱 [Cisco IOS交換服務命令參考指南](#)。

RkiosPortMan埠稽核進程中的CPU使用率較高

Catalyst 4500可能會在Cisco IOS軟體版本12.2(25)EWA和12.2(25)EWA1的 `show platformhealth` command的輸出中顯示執行 **RkiosPortMan Port Review** process時的CPU使用率較高。思科漏洞 [IDCSCeh08768](#) 會導致高使用率，Cisco IOS軟體版本12.2(25)EWA2可解決該問題。此程式為背景程式，不會影響Catalyst 4500交換器的穩定性。



附註：只有完成註冊的思科使用者能存取思科內部工具與資訊。

---

<#root>

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

使用中繼埠連線到IP電話時的CPU使用率較高

如果為語音VLAN選項和接入VLAN選項配置了埠，則該埠充當多VLAN接入埠。其優點是，只有為語音和接入VLAN選項配置的VLAN才會建立中繼。

中繼到電話的VLAN會增加STP例項的數量。交換機管理STP例項。對STP例項增加的管理，也會提高STP CPU利用率。

所有VLAN的中繼還會導致不必要的廣播、組播和未知的單播流量到達電話鏈路。

```
<#root>
```

```
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

使用RSPAN和第3層控制封包的CPU使用率高

使用RSPAN擷取的第3層控制封包是指定給CPU的，而不是只指定RSPAN目的地介面，這會造成CPU使用率高。L3控制資料包由具有轉發到CPU操作的靜態CAM條目捕獲。靜態CAM條目對所有VLAN都是全局性的。為了避免不必要的CPU泛洪，請使用Cisco IOS軟體版本12.2(37)SG和更新版本中提供的「每個VLAN控制流量攔截」功能。

<#root>

Switch(config)#

```
access-list hardware capture mode vlan
```

靜態ACL安裝在輸入功能TCAM的頂部，以捕獲發往224.0.0.\*範圍內公認IP組播地址的控制資料包。靜態ACL會在啟動時安裝，並出現在任何使用者配置的ACL之前。靜態ACL總是首先命中，並攔截發往所有VLAN上CPU的控制流量。

每個VLAN控制流量攔截功能提供用於捕獲控制流量的選擇性每個VLAN路徑管理模式。輸入特徵TCAM中的相應靜態CAM條目在新模式下無效。控制封包會透過附加至已啟用窺探或路由功能之VLAN的功能特定ACL來擷取。沒有功能特定ACL附加到RSPAN VLAN。因此，從RSPAN VLAN接收的所有第3層控制封包不會轉送到CPU。

## 用於分析發往CPU的流量的故障排除工具

如本文檔所示，發往CPU的流量是Catalyst 4500上CPU使用率較高的主要原因之一。發往CPU的資料流可能由於配置而有意產生，也可能由於配置錯誤或拒絕服務攻擊而無意產生。CPU具有內建QoS機制，可防止由於此流量而導致任何不良的網路影響。但是，請確定計算密集型資料流的根本原因，並在不需要時消除資料流。

### 工具1：使用SPAN監控CPU流量- Cisco IOS軟體版本12.1(19)EW及更高版本

Catalyst 4500允許使用標準SPAN功能監控進入CPU的流量（輸入或輸出）。目的地介面會連線到封包監控器或執行封包監聽器軟體的管理員膝上型電腦。此工具有助於快速準確地分析CPU處理的流量。該工具能夠監控繫結到CPU資料包引擎的單個隊列。

---

注意：交換引擎有32個CPU流量隊列，而CPU資料包引擎有16個隊列。

---



<#root>

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
```

```
-----
```

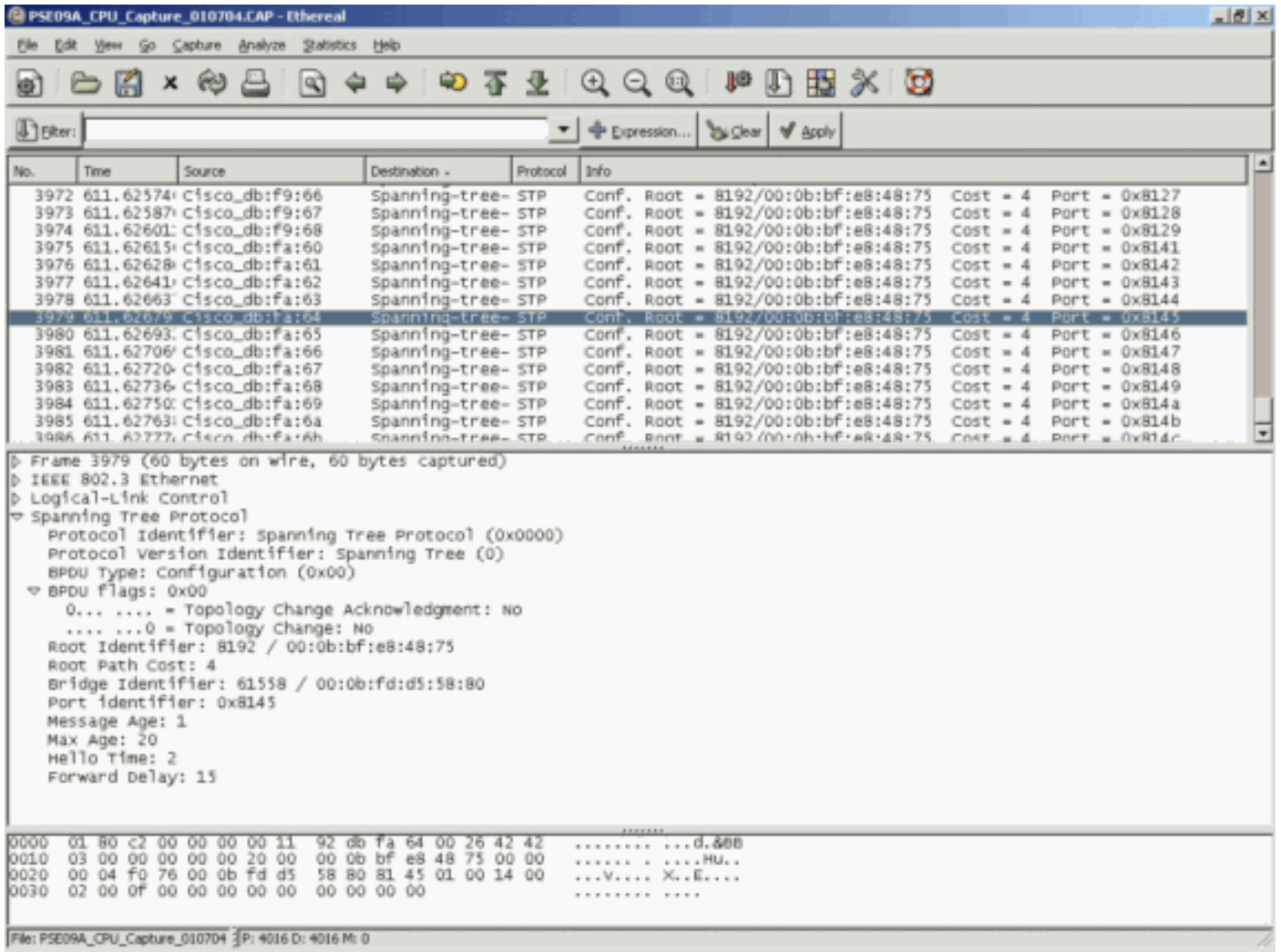
```
Type                : Local Session  
Source Ports        :  
    RX Only         : CPU  
Destination Ports   : Gi1/3  
    Encapsulation   : Native  
        Ingress     : Disabled  
        Learning    : Disabled
```

如果連線到執行監聽器程式的PC，便可快速分析流量。在本部分的窗口中顯示的輸出中，您可以看到CPU使用率高的原因是STP BPDUs數量過多。



**注意：** CPU嗅探器中的STP BPDU正常。但是，如果您看到的內容超出預期，則表明您的Supervisor引擎已超出建議限制。有關詳細資訊，請參閱本文檔的大量的生成樹埠例項部分。

---



CPU使用率過高是STP BPDU數量過多

工具2：內建CPU嗅探器- Cisco IOS軟體版本12.2(20)EW及更高版本

Catalyst 4500提供內建CPU嗅探器和解碼器，可快速辨識傳送到CPU的資料流。可以使用debug 命令啟用此工具，如本部分中的示例所示。此功能實現了一個循環緩衝區，一次可保留1024個資料包。當新資料包到達時，它們會覆蓋較舊的資料包。當您對高CPU使用率問題進行故障排除時，可以安全地使用此功能。

```
<#root>
```

```
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



註：發出debug 命令時，CPU使用率幾乎總是100%。發出debug 命令時，CPU使用率較高是正常的。

---

### 工具3：辨識將流量傳送到CPU的介面- Cisco IOS軟體版本12.2(20)EW及更高版本

Catalyst 4500提供了另一個有用的工具，用於確定為CPU處理傳送流量/資料包的頂級介面。此工具可幫助您快速辨識向CPU傳送大量廣播攻擊或其他拒絕服務攻擊的錯誤裝置。當您對高CPU使用率問題進行故障排除時，也可安全地使用此功能。

<#root>

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



注意：發出debug命令時，CPU使用率幾乎總是100%。發出debug命令時，CPU使用率較高是正常的。

---

## 摘要

Catalyst 4500交換器可在硬體中處理高速率的IP版本4 (IPv4)封包轉送。某些功能或例外狀況可能會導致透過CPU處理作業路徑轉送某些封包。Catalyst 4500使用複雜的QoS機制來處理計算密集型資料包。此機制確保了交換機的可靠性和穩定性，同時最大程度地提高了CPU用於資料包的軟體轉發。Cisco IOS軟體版本12.2(25)EWA2和更新版本針對封包/程式處理以及帳戶處理提供額外的增強功能。Catalyst 4500還具有足夠的命令和功能強大的工具，可幫助確定CPU使用率較高的問題的根本原因。但是，在大多數情況下，Catalyst 4500上的CPU使用率高不會導致網路不穩定，也不會引起擔憂。

## 相關資訊



- [執行CatOS軟體的Catalyst 4500/4000、2948G、2980G和4912G交換器上的CPU使用率](#)
- [LAN 產品支援頁面](#)
- [LAN 交換支援頁面](#)
- [技術支援與文件 - Cisco Systems](#)

## 關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。