

排除ASR1000系列路由器上的高CPU故障

目錄

[簡介](#)

[必備條件](#)

[需求](#)

[說明](#)

[疑難排解步驟](#)

[第1步 — 識別高CPU的模組](#)

[步驟2 — 分析模組](#)

[步驟3 - IOS進程](#)

[第4步 — Linux進程](#)

[第5步 — FECF流程](#)

[第6步 — QFP利用率](#)

[第7步 — 確定根本原因並確定修復](#)

[疑難排解範例](#)

[其他命令](#)

[路由處理器](#)

[嵌入式服務處理器](#)

簡介

本文檔介紹如何對ASR1000系列路由器上的高CPU問題進行故障排除。

必備條件

需求

思科建議您瞭解[ASR1000架構](#)以解釋和利用此文檔。

說明

Cisco路由器上的CPU使用率較高可定義為路由器上的CPU使用率高於正常使用率的條件。在某些情況下會出現CPU使用率增加，而在其他情況下則可能出現問題。可以忽略由於網路更改或配置更改導致的路由器上的臨時高CPU使用率，這是預期行為。

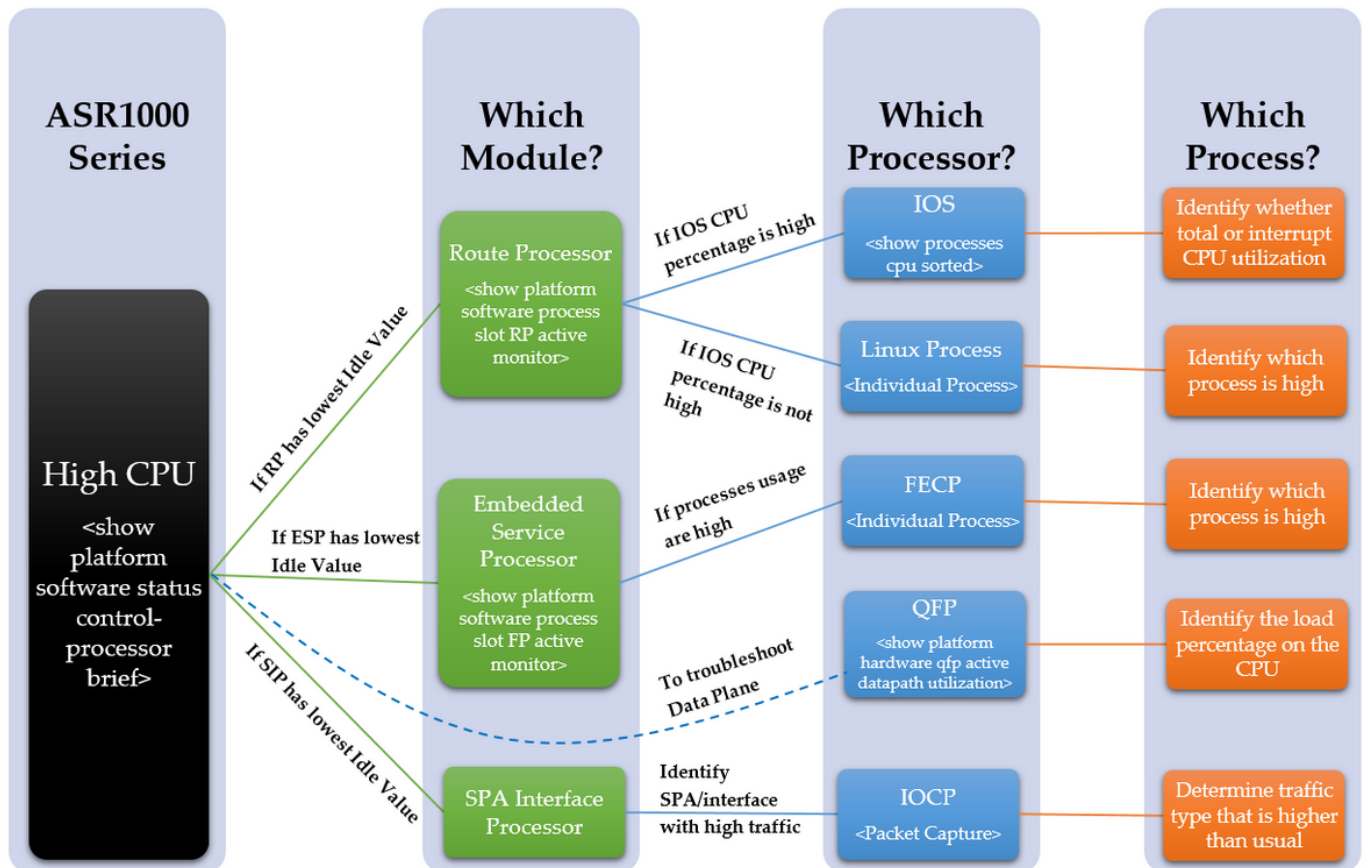
但是，如果路由器在較長時間內遇到較高的CPU使用率而不對網路或配置進行任何更改，這種情況是罕見的，需要進行分析。因此，當過度使用時，CPU無法主動為所有其他進程提供服務，這會導致命令列緩慢、控制平面延遲、資料包丟棄以及服務失敗。

CPU使用率高的原因包括：

1. 控制平面CPU接收過多的點傳流量
2. 一種進程，其行為出乎意料並導致CPU使用過度
3. 資料平面處理器過度使用/超額使用
4. 處理器中斷過多

CPU使用率高並不一定是ASR1000系列路由器問題，因為路由器CPU使用率與路由器上的負載成正比。例如，如果網路發生更改，則由於網路將重新收斂，這將導致大量控制平面流量。因此，我們需要確定CPU使用過度的根本原因，以確定它是預期行為還是問題。

下面是一個示意圖，詳細介紹如何解決高CPU問題的逐步過程：



疑難排解步驟

第1步 — 識別高CPU的模組

ASR1000在不同模組中有多個不同的CPU。因此，我們需要看到哪個模組顯示比正常使用率要高。這可以通過Idle值看到，因為空閒值越低，該模組的CPU利用率越高。這些不同的CPU都反映了模組的控制平面。

確定觀察裝置內哪個模組的CPU使用率較高。是使用以下命令的RP、ESP還是SIP

```
show platform software status control-processor brief
```

請參閱以下輸出以檢視突出顯示的列

如果RP的空閒值較低，則繼續執行步驟2第1點

如果ESP的空閒值較低，則繼續執行步驟3第2點

如果SIP的空閒值較低，則繼續執行步驟4第3點

```
Router#show platform software status control-processor brief
```

平均負載

插槽狀態1-Min 5-Min 15-Min

RP0正常0.00 0.02 0.00

ESP0正常0.01 0.02 0.00

SIP0正常0.00 0.01 0.00

記憶體(kB)

已用插槽狀態總計(Pct) 可用(Pct) 已提交(Pct)

RP0正常20093761879196率(94%)130180(6%)1432748(71%)

ESP0正常2009400692100率(34%)1317300率(66%)472536率(24%)

SIP0 Healthy 471804 284424(60%)187380(40%)193148(41%)

CPU利用率

插槽CPU使用者系統良好空閒IRQ SIRQ IOWait

RP0 0 2.59 2.49 0.00 **94.80** 0.00 0.09 0.00

ESP0 0 2.30 17.90 0.00 **79.80** 0.00 0.00 0.00

SIP0 0 1.29 4.19 0.00 **94.41** 0.09 0.00 0.00

如果Idle值都相對較高，則可能不是控制平面問題。要排除資料平面故障，需要觀察ESP的QFP。由於QFP過度使用（不會導致控制平面處理器上的高CPU），仍然可以觀察到「高CPU」的症狀。請繼續執行步驟6。

步驟2 — 分析模組

- 路由處理器

在RP內使用下面的命令確認觀察到哪個處理器具有高CPU利用率。是Linux進程還是IOS？

```
show platform software process slot RP active monitor
```

如果IOS CPU百分比高(linux_iosd-imag)，則是RP IOS。繼續執行步驟3

如果其他進程的CPU百分比很高，則可能是Linux進程。繼續執行步驟4

- 嵌入式服務處理器

在ESP中確認是否觀察到控制平面處理器的CPU使用率高。是FECF嗎？

```
show platform software process slot FP active monitor
```

如果進程高，則它是FECp，然後繼續執行步驟5

如果不是FECp，則它不是ESP中的控制平面進程相關問題。如果仍然觀察到網路延遲或隊列丟棄等症狀，則可能需要檢查資料平面是否存在過度利用。繼續執行步驟6

- SPA介面處理器

如果觀察到SIP具有高CPU利用率，則會發現IOCP具有高CPU。確定觀察到IOCP中的哪些進程具有較高的CPU利用率。

執行資料包捕獲，並確定哪些流量高於平常，以及哪些進程與此型別的流量相關聯。繼續執行步驟7

步驟3 - IOS進程

請參閱以下輸出，第一個百分比是總CPU使用率，第二個百分比是中斷CPU使用率，即用於處理打孔資料包的CPU使用率。

如果中斷百分比高，則表示大量流量被傳送到RP，(這可以通過show platform software infrastructure punt命令進行確認)

如果中斷百分比低，但總CPU數高，則會觀察到一個或多個進程在較長時間內使用CPU。

在IOS中，使用下面的命令確認觀察到哪些進程的CPU使用率高。

```
show processes cpu sorted
```

確定哪個百分比高（總CPU或中斷CPU），然後根據需要確定各個進程/進程。繼續執行第7步

```
Router#show processes cpu sorted
```

```
五秒內的 CPU 使用率:0% / 0%; 一分鐘:1%; 五分鐘:1%
```

```
PID運行時 ( 毫秒 ) 呼叫秒5秒1分鐘5分鐘TTY過程
```

```
PID運行時 ( 毫秒 ) 呼叫秒5秒1分鐘5分鐘TTY過程
```

```
188 8143 434758 18 0.15% 0.18% 0.19% 0 乙太網Msec Ti
```

```
515 380 7050 53 0.07% 0.00% 0.00% 0 SBC主流程
```

```
3 2154 215 10018 0.07% 0.00% 0.19% 0 Exec
```

```
380 1783 55002 32 0.07% 0.06% 0.06% 0 MMA DB計時器
```

```
63 3132 11143 281 0.07% 0.07% 0.07% 0 IOSD ipc任務
```

```
5 1 2 500 0.00% 0.00% 0.00% 0 IPC ISSU派單
```

```
6 19 12 1583 0.00% 0.00% 0.00% 0 RF從主Th
```

```
8 0 1 0 0.00% 0.00% 0.00% 0 RO通知計時器
```

```
7 0 1 0 0.00% 0.00% 0.00% 0 EDDRI_MAIN
```

```
10 6 75 80 0.00% 0.00% 0.00% 0池管理器
```

```
9 5671 538 10540 0.00% 0.14% 0.12% 0檢查堆
```

第4步 — Linux進程

如果觀察到IOS過度利用了CPU，則需要觀察單個linux進程的CPU利用率。這些進程是從show platform software process slot RP active monitor中列出的其他進程。確定觀察到哪些進程的CPU使用率較高，然後繼續執行步驟7。

第5步 — FECF流程

如果某個進程高，則很可能是FECF中負責高CPU利用率的進程，請繼續執行步驟7

第6步 — QFP利用率

量子流處理器是轉發ASIC。要確定轉發引擎上的負載，可以監控QFP。以下命令以資料包/秒、位/秒為單位列出輸入和輸出資料包（優先順序和非優先順序）。最後一行顯示由於資料包轉發而導致的CPU負載總量（百分比）。

```
show platform hardware qfp active datapath utilization
```

確定輸入或輸出是否較高，並檢視流程負載，然後繼續執行步驟7

```
Router#show platform hardware qfp active datapath utilization
```

```
CPP 0:子程式0 5秒1分5分60分
```

```
輸入：優先順序(pps)0 0 0
```

```
(bps)208 176 176 176
```

```
非優先順序(pps)0 2 2 2
```

```
(bps)64 784 784 784
```

```
總計(pps)0 2 2 2
```

```
(bps)272 960 960 960
```

```
輸出：優先順序(pps)0 0 0
```

```
(bps)192 160 160 160
```

```
非優先順序(pps)0 1 1
```

```
(bps)0 6488 6496 648
```

總計(pps)0 1 1

(bps)192 6648 6656 6648

正在處理：載入(pct)0 0 0

第7步 — 確定根本原因並確定修復

當發現某個或多個進程過度利用了CPU後，就可以更清楚地瞭解為什麼會發生CPU使用率過高的問題。要繼續，請研究由已識別的流程執行的功能。這將有助於確定如何解決該問題的行動計畫。例如 — 如果進程負責特定的協定，則可能需要檢視與此協定相關的配置。

如果您仍然遇到與CPU有關的問題，建議聯絡TAC，以便讓工程師幫助您進一步排除故障。上述故障排除步驟有助於工程師更有效地隔離問題。

疑難排解範例

在本例中，我們將運行故障排除過程，並嘗試最佳確定路由器高CPU的可能根本原因。首先，確定觀察到哪個模組遇到高CPU，我們得到以下輸出：

```
Router#show platform software status control-processor brief
```

平均負載

插槽狀態1-Min 5-Min 15-Min

RP0正常0.66 0.15 0.05

ESP0正常0.00 0.00 0.00

SIP0正常0.00 0.00 0.00

記憶體(kB)

已用插槽狀態總計(Pct) 可用(Pct) 已提交(Pct)

RP0正常20093761879196率(94%)130180(6%)1432756(71%)

ESP0正常2009400692472率(34%)1316928率(66%)472668率(24%)

SIP0 Healthy 471804 284556(60%)187248(40%)193148(41%)

CPU利用率

插槽CPU使用者系統Nice Idle IRQ SIRQ IOWait

RP0 0 57.11 14.42 0.00 **0.00** 28.25 0.19 0.00

ESP0 0 2.10 17.91 0.00 79.97 0.00 0.00 0.00

SIP0 0 1.20 6.00 0.00 92.80 0.00 0.00 0.00

由於RP0內的空間量非常低，這表明路由處理器內存在高CPU問題。因此，為了進一步排除故障，我們將確定RP中觀察到哪些處理器的CPU使用率較高。

```
Router#show processes cpu sorted
```

五秒內的 CPU 使用率:**84%** / **36%**;一分鐘:34%;五分鐘:9%

PID運行時 (毫秒) 呼叫秒5秒1分鐘5分鐘TTY過程

107 303230 50749 5975 46.69% 18.12% 4.45% 0 IOSXE-RP噴頭伺服器

63 105617 540091 195 0.23% 0.10% 0.08% 0 IOSD ipc任務

```

159 74792 2645991 28 0.15% 0.06% 0.06% 0 VRRS主執行緒
116 53685 169683 316 0.15% 0.05% 0.01% 0次工作
  9 305547 26511 11525 0.15% 0.28% 0.16% 0檢查堆
188 362507 20979154 17 0.15% 0.15% 0.19% 0 Ethernet Msec Ti
  3 147 186 790 0.07% 0.08% 0.02% 0 Exec
  2 32126 33935 946 0.07% 0.03% 0.00% 0負載計
446 416 33932 12 0.07% 0.00% 0.00% 0 VDC進程
164 59945 5261819 11 0.07% 0.04% 0.02% 0 IP ARP重試時間
 43 1703 16969 100 0.07% 0.00% 0.00% 0 IPC Keep Alive M

```

從輸出中可以看到，總CPU百分比和中斷百分比高於預期。利用CPU的頂級進程是「IOSXE-RP Punt Se」，這是處理RP CPU流量的進程，因此我們可以進一步檢視被推送到RP的流量。

Router#**show platform software infrastructure punt**

LSMPI介面內部統計資訊：

enabled=0, disabled=0, throttled=0, unthrottled=0, state is ready

輸入緩衝區= 90100722

輸出緩衝區= 100439

rxdone計數= 90100722

txdone計數= 100436

Rx no particletype count = 0

Tx no particletype count = 0

來自陰影計數的txbuf = 0

資料包無開始= 0

資料包末端= 0

點刪除統計資訊：

錯誤的版本0

錯誤型別0

具有功能標頭0

平台標題為0

功能標頭缺少0

通用報頭不匹配0

錯誤的總長度0

錯誤資料包長度0

錯誤的網路偏移量0

非點頭標題0

未知連結型別0

無swib 1

錯誤的ESS功能報頭0

無ESS功能0

無SSLVPN功能0

Punt For Us type unknown 0

點擲原因超出範圍0

IOSXE-RP丟棄資料包原因：

62210226 Layer2控制和舊式資料包

147 ARP請求或響應資料包

27801234 For-us資料包

84426 RP<->QFP keepalive資料包

6 Glean鄰接資料包

1647 For-us控制資料包

FOR_US控制IPv4協定統計資訊：

1647 OSPF資料包

資料包直方圖 (500位元組/bin) , 92中的平均大小 , 輸出56:

Pak-Size In-Count Out-Count

0+: 90097805 98790

500+: 0 7

從該輸出中，我們可以看到「For-us data packets」中有大量封包，這表示導向路由器的流量，此計數器確認已在數分鐘內多次透過觀察命令而遞增。這可以確定CPU被大量突發流量 (通常是控制平面流量) 過度利用。控制平面流量可能包括ARP、SSH、SNMP、路由更新(BGP、EIGRP、OSPF)等。根據此資訊，我們能夠確定CPU使用率高的潛在原因，這有助於排除根本原因。例如，可以實施資料包捕獲或不同流量的監控來檢視發往RP的確切流量，這樣就可以識別並解決根本原因，防止將來出現類似問題。

完成封包擷取後，可能輸入流量的一些範例如下：

- ARP:這可能是因為ARP請求數量過多，如果多個IP地址通過配置IP路由向廣播介面傳送ARP請求，就會發生這種情況。這也可能是由於從ARP表中清除的條目，並且必須根據過期的MAC地址條目或介面啟動/關閉重新獲取。
- SSH：這可能會導致高CPU，原因可能是大型show命令(show tech-support)或啟用了大量debug命令，這會強制通過SSH會話傳送大量CLI。
- SNMP:這可能是由於SNMP代理處理請求需要較長時間，從而導致高CPU。通常有兩個可能的原因是被輪詢的MIB，或由NMS輪詢的路由和/或ARP表。
- 路由更新：路由更新的大量湧入通常是由於網路重新收斂或鏈路抖動。這可能表示網路內已關閉的路由，或者整個裝置已關閉，這將迫使網路收斂並重新計算最佳路由，這取決於所使用的路由協定。

這突出表明，當進入單個進程級別時，如何通過識別高CPU的原因來隔離根本原因。在此處，可以隔離地分析各個進程或協定，以確定它是配置問題、軟體問題、網路設計還是預期實踐。

其他命令

下面列出了要使用的其他有用命令，並按它們所在的處理器排序：

路由處理器

- <show process cpu history>

- 提供過去60秒、分鐘和72小時的CPU歷史記錄圖表
- <show process process_ID>
 - 有關單個進程記憶體和CPU分配的詳細資訊
- <show platform software infrastructure punt>
 - 提供轉發到RP的所有流量的資訊
- <show platform software status control-processor brief>
 - 詳細描述CPU的負載和「運行狀況」，以及記憶體和模組統計資訊
- <show platform software process slot r0|r1 monitor>
 - 詳細介紹所選模組上的不同進程及其CPU和記憶體分配
- <monitor platform software process r0|r1>
 - 提供即時源，用於在進程使用CPU時對其進行更新
 - 要求首先在全域性配置模式下輸入「terminal terminal-type」命令才能正常工作

嵌入式服務處理器

- <show platform software process list fp active summary>
 - 詳細描述CPU上運行的所有進程以及平均負載的摘要
- <show platform software process slot f0|f1 monitor>
 - 詳細介紹所選模組上的不同進程及其CPU和記憶體分配
- <monitor platform software process f0|f1>
 - 提供在進程使用CPU時更新進程的即時源
 - 要求首先在全域性配置模式下輸入「terminal terminal-type」命令才能正常工作

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）。