

ASR 1000 シリーズ ルータ上の高い CPU 使用率のトラブルシューティング

目次

[概要](#)

[前提条件](#)

[要件](#)

[説明](#)

[トラブルシューティングの手順](#)

[ステップ 1-高CPU のモジュールを識別して下さい](#)

[ステップ 2-モジュールを分析して下さい](#)

[ステップ 3-IOSプロセス](#)

[ステップ 4-Linux プロセス](#)

[ステップ 5-FECP プロセス](#)

[ステップ 6-QFP 利用](#)

[ステップ 7-根本的な原因を判別し、修正を識別して下さい](#)

[例を解決して下さい](#)

[追加コマンド](#)

[ルート プロセッサ](#)

[組み込みサービス処理機構](#)

概要

この文書に ASR1000 シリーズ ルータの高CPU 問題を解決する方法を記述されています。

前提条件

要件

Cisco はこの資料を解読し、利用するために [ASR1000 アーキテクチャ](#)を理解することを推奨します。

説明

Ciscoルータの高CPU はルータの CPU稼働率が正常な使用方法の上にある条件と定義されるかもしれませんが。あるシナリオでは増大したCPU 使用法はもう1つのシナリオでは問題を示唆する可能性がある間、期待されます。ネットワーク変更かコンフィギュレーション変更によるルータの一時 CPU使用率が高い状態は無視することができ、予期された動作です。

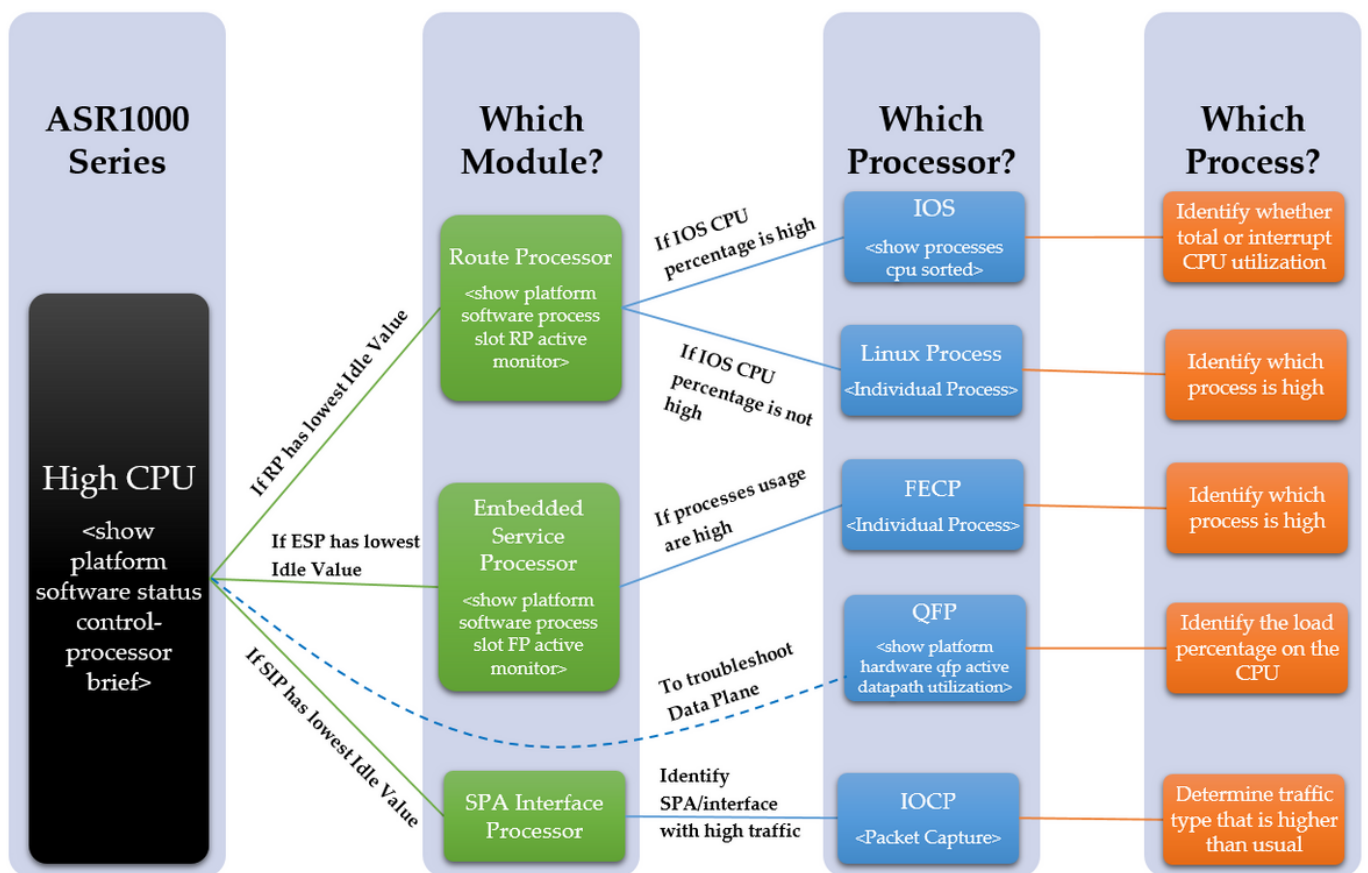
ただし、ネットワークまたは設定の変更なしで長期のための CPU使用率が高い状態を経験するルータは珍しく、分析される必要があります。従って過度に使用されたとき、サービスの遅いコマンド・ライン、コントロールプレーンレイテンシー、パケット破棄および失敗という結果に終るCPU はアクティブに他のプロセスをすべて保守できません。

高CPUの原因は次のとおりです:

1. コントロールプレーン CPU はたくさんのパケットトラフィックを受信します
2. プロセス予想に反して動作し、CPU 過度の使用という結果に終る
3. データ平らなプロセッサは過剰使用された /oversubscribed です
4. 余りにも多くのプロセッサの割り込み

高CPU はルータCPU利用率がルータのロードに正比例しているため ASR1000 シリーズ ルータ 問題常にではないです。たとえばネットワーク変更があれば、ネットワークが再収束するのでこれにより多量のコントロールプレーントラフィックを引き起こします。従ってそれが予期された動作または問題だったかどうか確認する、CPU 過度の使用の根本的な原因を判別する必要があります。

高CPU問題を解決する方法のステップバイステッププロセスを詳述するダイアグラムは下記にあります:



トラブルシューティングの手順

ステップ 1-高CPUのモジュールを識別して下さい

ASR1000 に複数異なるモジュールを渡る別の CPU があります。従って、どのモジュールが非常により正常な使用方法を表示するか参照する必要があります。これはアイドル状態の値によって、ように下部のアイドル状態の値見られる場合があります、より高いそのモジュールの CPU稼働率。これらの異なる CPU はすべてモジュールのコントロールプレーンを反映します。

高CPUを経験するためにデバイス内のどのモジュールが観察されるか判別して下さい。それは下記のコマンドで RP、ESP、または SIP あります

Control Processor

強調表示されたカラムを表示するために下記の出力を参照して下さい

RP に低いアイドル状態の値がある場合、ステップ 2 ポイント 1 に進んで下さい

ESP に低いアイドル状態の値がある場合、ステップ 3 ポイント 2 に進んで下さい

SIP に低いアイドル状態の値がある場合、ステップ 4 ポイント 3 に進んで下さい

```
Router#show Control Processor
```

```
 1  5  15
RP0  0.00 0.02 0.00
ESP0  0.01 0.02 0.00
SIP0  0.00 0.01 0.00

kb
PctPctPct
RP0  2009376 1879196 94% 130180 6% 1432748 71%
ESP0  2009400 692100 34% 1317300 66% 472536 24%
SIP0  471804 284424 60% 187380 40% 193148 41%

CPU
CPU    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
```

アイドル状態の値がすべての比較的高い場合、それはコントロールプレーン問題ではないかもしれませんが。データを解決するために ESP の QFP を観察される必要があります平にして下さい。「高CPU」の現象はまだコントロールプレーンプロセッサの高CPUという結果に終らない過剰使用された QFP が観察された原因である場合もあります。ステップ 6 に進みます。

ステップ 2 –モジュールを分析して下さい

• ルートプロセッサ

下記のコマンドで CPU使用率が高い状態があるためにプロセッサは観察される RP の内で確認して下さい。それは Linux プロセスまたは IOS ですか。

```
RP
```

IOS CPU パーセントが高ければ (linux_iosd-imag)、です RP IOS あります。STEP 3 に進んで下さい

他のプロセスの CPU パーセントが高い場合、Linux プロセスであることは可能性が高いです。STEP 4 に進んで下さい

• 組み込みサービス処理機構

CPU使用率が高い状態があるためにコントロールプレーンプロセッサは観察されるかどうか ESP の内で確認して下さい。それは FECF ですか。

プロセスが高ければそれは FECP、STEP 5 に進みます

FECP ではない場合、それは ESP 内のコントロールプレーン プロセス関連 問題ではないです。ネットワーク 待ち時間のような現象かキュー ドロップがそれでも観察される場合、データ平面は過度の使用のために検討される必要がある場合もあります。STEP 6 に進んで下さい

• SPA インターフェイス プロセッサ

CPU使用率が高い状態をもらうように SIP がそして IOCP 観察されたら高CPU を持つために観察されて下さい。CPU使用率が高い状態があるために IOCP 内のプロセスかプロセスは観察されるかどれ判別して下さい。

パケットキャプチャを行い、どのトラフィックが通常より高く、どのプロセスがこのトラフィックの種類と関連付けられるか特定して下さい。STEP 7 に進んで下さい

ステップ 3 – IOSプロセス

下記の出力を参照して下さい、最初のパーセントは CPU合計使用率であり、パントされたパケットを処理するのに使用される CPU の量の第 2 パーセントは割り込み CPU稼働率です。

割り込みパーセントが高ければ多量のトラフィックが RP にパントされること示しますコマンドで、(これを確認することができます示しますプラットフォームソフトウェア インフラストラクチャ パントを)

割り込みパーセントが低い、総 CPU が高い場合、CPU を長期間にわたって利用するために観察されるプロセスまたはプロセスがあります。

下記のコマンドで CPU使用率が高い状態があるためにプロセスかプロセスは観察される IOS の内で確認して下さい。

```
show processes cpu
```

どのパーセントが高い (総 CPU はまたは CPU を割り込みます) 識別し、次に必要であれば個々のプロセス/プロセスをか特定して下さい。STEP 7 に進んで下さい

```
Router#show CPU
```

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

```
PID Runtimems uSecs 5Sec 1Min 5Min TTY Process
```

```
PID Runtimems uSecs 5Sec 1Min 5Min TTY Process
```

```
188 8143 434758 18 0.15% 0.18% 0.19% 0 Ti
```

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

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

```

380 1783 55002 32 0.07% 0.06% 0.06% 0 MUTTAHIDA MAJLIS-E-AMAL DB
63 3132 11143 281 0.07% 0.07% 0.07% 0 IOSD IPC
5 1 2 500 0.00% 0.00% 0.00% 0 IPC ISSU Dispatc
6 19 12 1583 0.00% 0.00% 0.00% 0 RF Th
8 0 1 0 0.00% 0.00% 0.00% 0 RO
7 0 1 0 0.00% 0.00% 0.00% 0 EDDRI_MAIN
10 6 75 80 0.00% 0.00% 0.00% 0
9 5671 538 10540 0.00% 0.14% 0.12% 0

```

ステップ 4 – Linux プロセス

IOS が CPU を過剰使用するために観察される場合個々の Linux プロセスのために CPU稼働率を観察する必要があります。これらのプロセスは提示プラットフォームソフトウェア プロセス スロット RP アクティブ モニタからリストされている他のプロセスです。識別して下さい高CPUを経験するためにプロセスがプロセスが観察されたりそして STEP 7.に進む。

ステップ 5 – FECP プロセス

プロセスがプロセスが高かったらそれからそれ本当らしかったらそれら CPU使用率が高い状態に責任がある FECP 内のプロセス、STEP 7 に進んで下さい

ステップ 6 – QFP 利用

クオンタム フロー プロセッサは転送 ASIC です。フォワーディングエンジンのロードを判別するために、QFP は監視することができます。下記のコマンド リスト パケット毎秒の入出力パケット (優先順位および非優先)、およびビット/秒。最終的なライン表示パーセントのパケット転送による CPU負荷の総量。

```
qfp datapath
```

入力か出力が高かったら識別し、プロセス負荷を表示し、次に STEP 7 に進んで下さい

```

Router#show  qfp  datapath

CPP 0: Subdev 0 5 1 5 60

Input :  pps 0 0 0 0

        / 208 176 176 176

        pps 0 2 2 2

        / 64 784 784 784

```

```

pps 0 2 2 2
    / 272 960 960 960

pps 0 0 0 0
    / 192 160 160 160

pps 0 1 1 1
    / 0 6488 6496 6488

pps 0 1 1 1
    / 192 6648 6656 6648

: pct 0 0 0 0

```

ステップ7-根本的な原因を判別し、修正を識別して下さい

識別される CPU を過剰使用するために観察されるプロセスがプロセスを使ってクリア ピクチャがある高CPU がなぜの発生したか。 続行するために、特定されたプロセスによって実行された機能を研究して下さい。 これは問題にアプローチする方法のアクションプランを判別することを助けます。 たとえば-プロセスが特別なプロトコルに責任があればこのプロトコルに関する設定を検知したいと思う場合もあります。

それでも CPU 関連 問題に直面する場合、エンジニアが更に解決するのを助けることを可能にするように TAC に接触することを推奨します。 解決するべき上記のステップはエンジニアが問題をより効率的に特定するのを助けます。

例を解決して下さい

この例でプロセスによって解決するために動作し、推奨に試みるためにルータの高CPU のための可能性のある 根本的な原因を特定して下さい。 高CPU を経験するためにどのモジュールが観察されるか始まるために、私達持っています下記の出力を判別して下さい:

```

Router#show Control Processor

 1 5 15
RP0 0.66 0.15 0.05
ESP0 0.00 0.00 0.00
SIP0 0.00 0.00 0.00

kb
PctPctPct
RP0 2009376 1879196 94% 130180 6% 1432756 71%
ESP0 2009400 692472 34% 1316928 66% 472668 24%
SIP0 471804 284556 60% 187248 40% 193148 41%

CPU
CPU IRQ SIRQ IOwait

```

```
RPO 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
```

RP0 内のアイドル状態の量は非常に低いので、ルートプロセッサ内の高CPU 問題を提案します。従って高CPU を経験するために RP 内のどのプロセッサが観察されるか更に解決するために識別します。

```
Router#show CPU
CPU utilization for five seconds: 84%/36%; one minute: 34%; five
minutes: 9%
PID Runtimems uSecs 5Sec 1Min 5Min TTY Process
107 303230 50749 5975 46.69% 18.12% 4.45% 0 IOSXE-RP Se
63 105617 540091 195 0.23% 0.10% 0.08% 0 IOSD IPC
159 74792 2645991 28 0.15% 0.06% 0.06% 0 VRRS
116 53685 169683 316 0.15% 0.05% 0.01% 0
9 305547 26511 11525 0.15% 0.28% 0.16% 0
188 362507 20979154 17 0.15% 0.15% 0.19% 0 Ti
3 147 186 790 0.07% 0.08% 0.02% 0 Exec
2 32126 33935 946 0.07% 0.03% 0.00% 0
446 416 33932 12 0.07% 0.00% 0.00% 0 VDC
164 59945 5261819 11 0.07% 0.04% 0.02% 0 IP ARP
43 1703 16969 100 0.07% 0.00% 0.00% 0 IPC M
```

この出力から、それは総 CPU パーセントおよび割り込みパーセントが予想を越えたであること観察することができます。上プロセスはプロセスである「IOSXE-RP パント Se です」従って CPU を利用する RP CPU のためのトラフィックを処理する、RP にパントされるこのトラフィック更に調べることができる。

```
Router#show
LSMPI :
enabled=0 disabled=0throttled=0unthrottled=0
= 90100722
= 100439
rxdone = 90100722
txdone = 100436
Rx particletype = 0
Tx particletype = 0
Txbuf = 0
= 0
= 0
:
0
0
0
0
0
0
0
0
0
0
```

```

0
0
swidb 1
ESS 0
ESS 0
SSLVPN 0
0
0
IOSXE-RP :
62210226Layer2
147 ARP
27801234
84426 RP<->QFP
6
1647

FOR_US IPv4 protcol :
1647 OSPF
histogram(500 /92 avg 56:

0+: 90097805 98790
500+: 0 7

```

この出力から、数分にわたるコマンド複数回の観測からあるためにルータの方に送信されるトラフィックを示すに多量のパケットが「のため私達データパケット」このカウンター確認された増分したあることを見る場合があります。これは CPU が頻繁にコントロールプレーントラフィックである多量のパケットされたトラフィックによって過剰使用されることを確認します。コントロールプレーントラフィックは ARP、SSH、SNMP、ルート更新 (BGP、EIGRP、OSPF) 先祖などを含むことができます この情報から、高CPUの潜在的な原因を特定でき、これは根本的な原因のために解決することを助けます。同じような問題を将来防ぐために根本的な原因が特定され、解決するようにする RP にパケットされた正確なトラフィックを見るためにたとえば、別のトラフィックのパケットキャプチャがモニタは設定できます。

パケットキャプチャが完了すれば、可能性によってパケットされるトラフィックの例は次のとおりです:

- **ARP:** これは複数の IP アドレスがブロードキャストインターフェイスへ IP ルートの設定によって ARP 要求を送信 することだった場合発生する過剰 ARP 要求が原因である可能性があります。これはまた ARP テーブルからのフラッシュされたエントリが原因である可能性があります、ために基づいているかことエージング アウトするは MAC アドレスエントリに、またはインターフェイスさせますアップする/学び直されなければなりません。
- **SSH:** または多くの debug コマンドがイネーブルになっているとき多くの CLI を SSH セッションに送信 されるために強制するこれにより大きい表示コマンド (show tech-support) による高CPU を引き起こす可能性があります。
- **SNMP :** 従ってこれは要求を処理する時間の長時間を SNMP エージェントが原因である奪取する可能性があります高CPU を引き起こします。頻繁に 2 つの推定 原因はポーリングされる、またはルーティングしますおよび/または NMS によってポーリングされる ARP テーブルです MIB。
- **ルート更新:** 多くの場合ルート更新の流入はネットワーク再収束が原因です、またはリンクは

フラップします。これはどの強制をコンバートし、最良のルートをか計算し直すどのルーティングプロトコルがによって使用中であるか決まるネットワーク ネットワークの内で行く、またはたどって行く全体のデバイスを可能性がりますルーティングを示す。

これは個々のプロセスレベルに来るとき根本的な原因が高CPUの原因の識別を通してどのように特定することができるか強調表示します。ここから、個々のプロセスかプロトコルは分離でそれが設定に関する問題、ソフトウェア上の問題、ネットワーク設計、または意図されていた推奨事項であるかどうか識別するために分析することができます。

追加コマンド

下記は利用するプロセッサによってに関連している他の追加役に立つコマンドのリストで、ソートされます:

ルート プロセッサ

- **<show プロセス CPU history>** 最後の 60 秒、分および 72 時間にわたる CPU 履歴のグラフを提供します
- **<show プロセス process_ID>** 個々のプロセス メモリおよび CPU アロケーションの詳細な情報
- **<show プラットフォームソフトウェア インフラストラクチャ punt>** RP にパントされるすべてのトラフィックで情報を提供します
- **<show プラットフォームソフトウェア ステータス Control Processor brief>** CPU のロードおよび「健全性を詳述しましたり、またメモリおよびモジュール統計情報を詳述します
- **<show プラットフォームソフトウェア プロセス スロット r0|r1 monitor>** 選択したモジュールの異なるプロセスおよび CPU およびメモリ 割り当て詳述します
- **<monitor プラットフォームソフトウェア プロセス r0|r1>** それらが CPU を利用するようにプロセスの更新ライブ映像を提供します正しく機能するためにグローバル コンフィギュレーション モードで最初に入力されるコマンド「ターミナル ターミナル型」を必要とします

組み込みサービス処理機構

- **<show プラットフォームソフトウェア プロセス リスト fp アクティブ summary>** CPU で動作する、また平均負荷詳述しますすべてのプロセスの要約を
- **<show プラットフォームソフトウェア プロセス スロット f0|f1 monitor>** 選択したモジュールの異なるプロセスおよび CPU およびメモリ 割り当て詳述します
- **<monitor プラットフォームソフトウェア プロセス f0|f1>** であるのでプロセスの更新が CPU を利用することライブ映像を提供します正しく機能するためにグローバル コンフィギュレーション モードで最初に入力されるコマンド「ターミナル ターミナル型」を必要とします