

Catalyst 4500スイッチでCPU使用率が高い場合のトラブルシューティング

内容

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[背景説明](#)

[Catalyst 4500 CPU パケット処理アーキテクチャについて](#)

[Catalyst 4500 の高 CPU 使用率に関する理由を特定する](#)

[基準 CPU 使用状況](#)

[Catalyst 4500 スイッチでの show processes cpu コマンドについて](#)

[Catalyst 4500 スイッチでの show platform health コマンドについて](#)

[一般的な CPU 高使用率の問題に関するトラブルシューティング](#)

[プロセス交換パケットによる高 CPU 使用率](#)

[スパニング ツリー ポート インスタンスの数が多](#)

[ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。](#)

[ステップ3:CPUに送られるトラフィックのタイプを特定するために、トラフィックを受信するCPUキューを確認します。](#)

[手順4：根本原因を特定します。](#)

[ICMPリダイレクト、同じインターフェイス上のルーティングパケット](#)

[ステップ1:show processes cpuコマンドを使用して、Cisco IOSプロセスを確認します。](#)

[ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。](#)

[ステップ3:CPUに送られるトラフィックのタイプを特定するために、トラフィックを受信するCPUキューを確認します。](#)

[手順4：根本原因を特定します。](#)

[IPX または AppleTalk ルーティング](#)

[ステップ1:show processes cpuコマンドを使用して、Cisco IOSプロセスを確認します。](#)

[ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。](#)

[ステップ3：トラフィックを受信するCPUキューを確認し、CPUに送られるトラフィックのタイプを特定します。](#)

[手順4：根本原因を特定します。](#)

[ホスト学習](#)

[ステップ1:show processes cpuコマンドを使用して、Cisco IOSプロセスを確認します。](#)

[ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。](#)

[ステップ3：トラフィックを受信するCPUキューを確認し、CPUに送られるトラフィックのタイプを特定します。](#)

[手順4：根本原因を特定します。](#)

[セキュリティ ACL 用のハードウェア リソース \(TCAM \) の枯渇](#)

[ステップ1:show processes cpuコマンドを使用して、Cisco IOSプロセスを確認します。](#)

[ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。](#)

[ステップ3:トラフィックを受信するCPUキューを確認し、CPUに送られるトラフィックのタイプを特定します。](#)

[ステップ4:問題を解決します。](#)

[ACL 内の log キーワード](#)

[ステップ1:show processes cpuコマンドを使用して、Cisco IOSプロセスを確認します。](#)

[ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。](#)

[ステップ3:トラフィックを受信するCPUキューを確認し、CPUに送られるトラフィックのタイプを特定します。](#)

[ステップ4:問題を解決します。](#)

[レイヤ2 転送ループ](#)

[ステップ1:show processes cpuコマンドでCisco IOSプロセスを確認します](#)

[ステップ2:show platform healthコマンドでCatalyst 4500固有のプロセスを確認します](#)

[ステップ3:CPUに送られるトラフィックのタイプを特定するために、トラフィックを受信するCPUキューを確認します](#)

[ステップ4:根本原因を特定し、問題を修正する](#)

[ステップ5:高度なSTP機能の実装](#)

[CPU の使用率が高くなるその他の原因](#)

[過剰なリンクフラップ](#)

[ステップ1:show processes cpuコマンドを使用して、Cisco IOSプロセスを確認します。](#)

[ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。](#)

[手順3:根本原因を特定します。](#)

[FIBの一貫性チェックによって CPU 使用率が一時的に高くなる](#)

[K2FibAdjMan Host Move プロセスでの高い CPU 使用率](#)

[RkiosPortMan Port Review プロセスにおける高い CPU 使用率](#)

[トランク ポートの使用で IP Phone に接続されたときの高い CPU 使用率](#)

[RSPAN とレイヤ 3 コントロール パケットを伴う高い CPU 使用率](#)

[CPU 宛のトラフィックを分析するためのツールのトラブルシューティング](#)

[ツール1:SPANでCPUトラフィックを監視する: Cisco IOSソフトウェアリリース12.1\(19\)EW以降](#)

[ツール2:組み込みCPUスニファ: Cisco IOSソフトウェアリリース12.2\(20\)EW以降](#)

[ツール3:トラフィックをCPUに送信するインターフェイスの特定: Cisco IOSソフトウェアリリース12.2\(20\)EW以降](#)

[要約](#)

[関連情報](#)

はじめに

このドキュメントでは、CPUのパケット処理アーキテクチャについて説明し、Catalyst 4500スイッチでの高CPU使用率の原因を特定する方法を示します。

前提条件

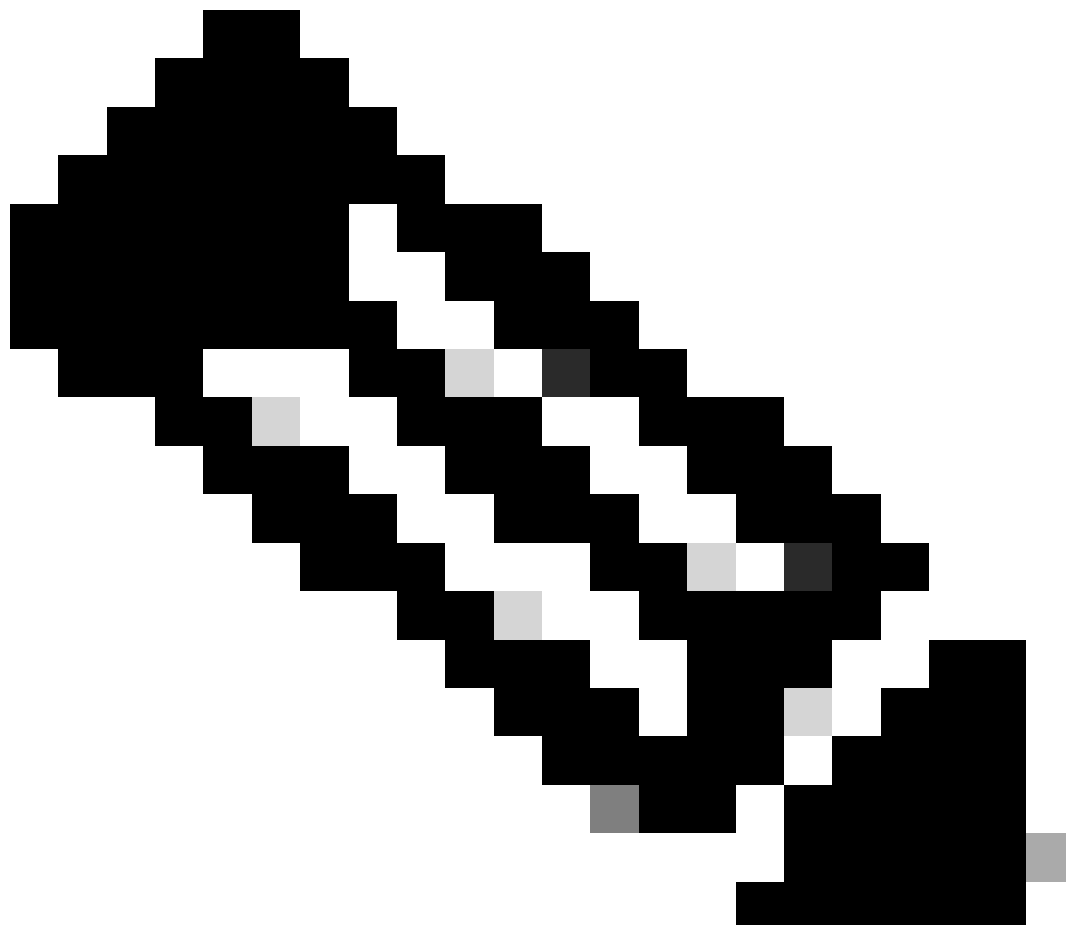
要件

このドキュメントに関する固有の要件はありません。

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- Catalyst 4500 シリーズ スイッチ
 - Catalyst 4948 シリーズ スイッチ
-



注：このドキュメントは、Cisco IOS®ソフトウェアベースのスイッチにのみ適用されません。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

表記法

ドキュメント表記の詳細については、『シスコテクニカルティップスの表記法』を参照してください。

背景説明

Catalyst 4500 シリーズ スイッチ (Catalyst 4948 スイッチを含む) には、CPU バウンド トラフィック向けの洗練されたパケット処理手法が組み込まれています。これらのスイッチでよく見られる問題は、高い CPU 使用率です。このドキュメントでは、CPU パケット処理アーキテクチャについて詳しく説明し、これらのスイッチでの高い CPU 使用率の原因を特定する方法について説明します。また、Catalyst 4500 シリーズで CPU 使用率が高くなる原因となる一般的なネットワークや設定のシナリオについても説明します

CPU のパケット処理アーキテクチャと CPU の使用率が高くなる問題のトラブルシューティングについて考える前に、ハードウェア ベースの転送を行うスイッチと Cisco IOS ソフトウェア ベースのルータでは CPU の使用方法が異なっていることを理解しておく必要があります。CPU の使用率が高くなる現象は、デバイス上のリソースの枯渇や、クラッシュの危険を示すものと誤解されていることが多いようです。容量の問題は、Cisco IOS ルータで CPU 使用率が高くなった場合に発生する症状の 1 つです。ところが、ハードウェアベースの転送を行う Catalyst 4500 のようなスイッチでは、容量の問題が CPU 高使用率の症状となることはほとんどありません。Catalyst 4500 は、ハードウェア Application-Specific Integrated Circuit (ASIC; 特定用途向け集積回路) でパケットを転送し、最大 1 億 200 万パケット/秒 (Mpps) のトラフィック転送速度に到達するように設計されています。

Catalyst 4500 の CPU は次の機能を実行します。

- 次のような設定済みのソフトウェア プロトコルを管理します。
 - スパニング ツリー プロトコル (STP)
 - ルーティング プロトコル
 - シスコ検出プロトコル (CDP)
 - Port Aggregation Protocol (PAgP)
 - VLAN Trunk Protocol (VTP; VLAN トランク プロトコル)
 - Dynamic Trunking Protocol (DTP; ダイナミック トランキング プロトコル)
- 次のようなハードウェア ASIC への設定/ダイナミック エントリをプログラムします。
 - Access Control List (ACL; アクセス コントロール リスト)
 - CEF エントリ
- 次のようなさまざまなコンポーネントを内部で管理します。
 - Power over Ethernet (PoE) ライン カード
 - 電源モジュール
 - ファントレイ

- 次のようにスイッチへのアクセスを管理します。
 - Telnet
 - コンソール
 - Simple Network Management Protocol (SNMP)
- 次のように、ソフトウェア パス経由でパケットを転送します。
 - Internetwork Packet Exchange (IPX) でルーティングされるパケット、これはソフトウェア パスでだけサポートされる
 - Maximum Transmission Unit (MTU; 最大伝送ユニット) フラグメント化

このリストによると、高い CPU 使用率は、CPU によるパケットの受信または処理の結果、発生する可能性があります。処理のために送信されるパケットの中には、ネットワーク操作に不可欠なものもあります。これらの基本パケットの例として、スパニング ツリートポロジ設定の Bridge Protocol Data Unit (BPDU; ブリッジ プロトコル データ ユニット) があります。ただし、他のパケットはソフトウェアで転送されるデータ トラフィックになる可能性があります。次に示すシナリオでは、パケットを処理するために CPU に送るスイッチング ASIC が必要になります。

- CPU にコピーされるものの、本来のパケットはハードウェアでスイッチされるパケット。

その例は、ホスト MAC アドレス学習です。

- 処理のために CPU に送信されるパケット

次に例を示します。

- ルーティング プロトコルの更新
- BPDU
- 意図的または意図的ではないトラフィックのフラッディング

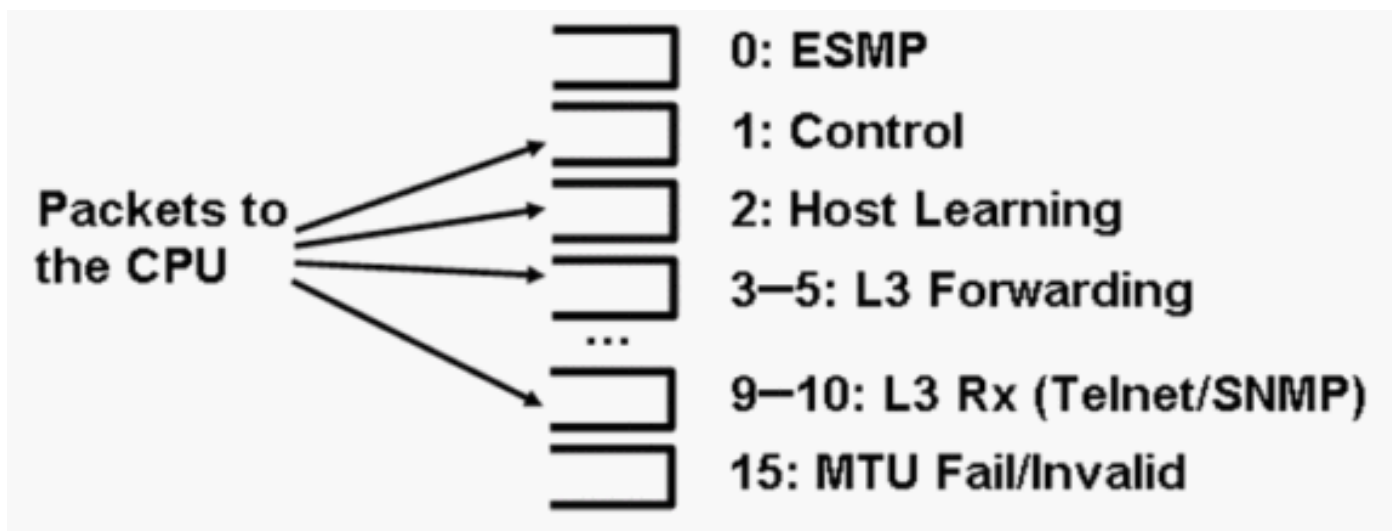
- 転送のために CPU に送信されるパケット

その例として、IPX または AppleTalk ルーティングを必要とするパケットがあります。

Catalyst 4500 CPU パケット処理アーキテクチャについて

Catalyst 4500 には、CPU を宛先とするトラフィックのタイプ間を差別化するための、組み込みの Quality of Service (QoS) メカニズムがあります。このメカニズムは、レイヤ2(L2)/レイヤ3(L3)/レイヤ4(L4)のパケット情報に基づいて差別化を行います。スーパーバイザ パケット エンジンには、さまざまなタイプのパケットやイベントを処理するために 16 個のキューがあります。[図 1 に、これらのキューを示します。表 1 は、キューと、各キューに入れられるパケットの種類を示しています。](#) Catalyst 4500 は、この 16 個のキューで、パケット タイプまたは優先度に基づいてパケットをキューイングできます。

図 1 : Catalyst 4500 は複数の CPU キューを使用する



Catalyst 4500は複数のCPUキューを使用します

表 1 : Catalyst 4500 キューの説明

キュー番号	キュー名	キューイングされるパケット
0	ESMP	ラインカードASICまたはその他のコンポーネント管理用のESMP ¹ パケット (内部管理パケット)
1	Control	STP、CDP、PAgP、LACP ² 、UDLD ³ などの L2 コントロール プレーン パケット
2	ホスト学習	L2 転送テーブルを構築するために CPU にコピーされる未知の発信元 MAC アドレスを伴うフレーム
3、4、5	L3 Fwd Highest,L3 Fwd High/Medium,L3 Fwd Low	GRE ⁴ トンネルなど、ソフトウェアで転送する必要があるパケット。宛先IPアドレスのARP ⁵ が解決されていない場合、パケットはこのキューに送信されます。
6、7、8	L2 Fwd Highest,L2 Fwd High/Medium,L2 Fwd Low	ブリッジングの結果として転送されるパケット <ul style="list-style-type: none"> • IPX や AppleTalk にルーティングされるパケットなど、ハードウェアでサポートされていないプロトコルが CPU にブリッジされます。 • ARP 要求と応答 • スイッチの SVI⁶/L3 インターフェイスの宛先 MAC アドレスを持つパケットは、次の理由によってハードウェアでルーティングできない場合にはブリッジされます。 <ul style="list-style-type: none"> ◦ IP ヘッダー オプション ◦ 期限切れになった TTL⁷ ◦ ARPA 以外のカプセル化
9 10	L3 Rx高、 L3 Rx低	CPU IPアドレスを宛先とするL3コントロールプレ

		ートラフィック (ルーティングプロトコルなど)。例としては、Telnet、SNMP、SSH ⁸ などがあります。
11	RPF 障害	RPF ⁹ チェックで失敗したマルチキャストパケット
12	ACL fwd(snooping)	DHCP ¹⁰ スヌーピング、ダイナミックARPインスペクション、またはIGMP ¹¹ スヌーピング機能によって処理されるパケット
13	ACL log, unreachable	キーワード ^{the} logkeywordでACE ¹² をヒットしたパケット、または出力ACLでの拒否や宛先へのルートの欠落によって廃棄されたパケット。これらのパケットには、ICMP到達不能メッセージを生成する必要があります。
14	ACL sw processing	セキュリティ ACL に対する TCAM ¹³ など、追加 ACL ハードウェア リソースの欠落のためにパントされるパケット
15	MTU Fail/Invalid	出カインターフェイス MTU サイズがパケットのサイズよりも小さいので、フラグメント化する必要のあるパケット

¹ESMP = Even Simple Management Protocol (シンプル管理プロトコル)。

²LACP = Link Aggregation Control Protocol (リンク集約制御プロトコル)。

³UDLD = UniDirectional Link Detection (単方向リンク検出)。

⁴GRE = generic routing encapsulation (総称ルーティングカプセル化)。

⁵ARP = Address Resolution Protocol (アドレス解決プロトコル)。

⁶SVI = switched virtual interface (スイッチ仮想インターフェイス)。

⁷TTL = 存続可能時間。

⁸SSH = Secure Shell Protocol (セキュアシェル)。

⁹RPF = リバースパス転送

¹⁰DHCP = Dynamic Host Configuration Protocol (ダイナミックホストコンフィギュレーションプロトコル)。

¹¹IGMP = Internet Group Management Protocol (インターネットグループ管理プロトコル)。

¹²ACE = アクセスコントロールエントリ。

¹³TCAM = ternary content addressable memory (三値連想メモリ)。

次のキューは独立したキューです。

- L2 Fwd Highest or L3 Fwd Highest

- L2 Fwd高/中位L3 Fwd高/中
- L2 FwdローまたはL3 Fwdロー
- L3 Rx高またはL3 Rx低

パケットは QoS ラベルに基づいてこれらのキューにキューイングされます。QoS ラベルとは、IP Type of Service (ToS; タイプ オブ サービス) からの Differentiated Services Code Point (DSCP; DiffServ コード ポイント) 値です。たとえば、DSCPが63のパケットは、L3 Fwd Highestqueueにキューイングされます。これらの16個のキューに対して受信または廃棄されるパケットは、show platform cpu packet statistics allcommandの出力で確認できます。このコマンドの出力は非常に長くなっています。ゼロ以外のイベントだけを表示するには、show platform cpu packet statisticsコマンドを発行します。代替コマンドは show platform cpuport コマンドです。show platform cpuportコマンドは、Cisco IOSソフトウェアリリース12.1(11)EW以前を使用している場合にだけ、使用してください。このコマンドは廃止されています。ただし、この古いコマンドはCisco IOSソフトウェアリリース12.2(20)EWAよりも前のCisco IOSソフトウェアリリースではshow tech-supportコマンドの一部でした。

すべてのトラブルシューティングにはshow platform cpu packet statisticsコマンドを使用します。

<#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
-------	-------	-----------	-----------	-----------	------------

Esm	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のCPUでは、[表1](#)に示すさまざまなキューに対して、[重み付け](#)を割り当てています。CPUは、[重要度](#)や[タイプ](#)を基準にして、また、[トラフィック優先度](#)や[DSCP](#)を基準にして重み付けをします。CPUは、キューの相対的な重みを基準にしてキューにサービスを提供します。たとえば、BPDUのように両方がコントロールパケットであり、ICMPエコー要求が待ち状態である場合、CPUは先にコントロールパケットにサービスを提供します。低優先度のトラフィックまたは重要度の低いトラフィックは、大量にあってもシステムの処理または管理を行うCPUの能力を不足させることはありません。このメカニズムによって、CPUの使用率が高い状態でもネットワークの安定性が確保されます。このネットワークの安定性を確保する仕組みは、理解しておく必要のある重要な事項です。

Catalyst 4500のCPUのパケット処理については、重要な事柄がもう1つあります。CPUが高優先度のパケットやプロセスにすでにサービスを提供しているにもかかわらず、特定の期間、余分なCPUサイクルがある場合、CPUは低優先度のキューパケットにサービスを提供したり、より低い優先度のバックグラウンドプロセスを実行したりします。低優先度のパケット処理やバックグラウンドプロセスの結果としての高CPU使用率は、CPUが常にすべての時間を利用可能にしようとするので、通常のことと見なされます。このように、CPUは、スイッチの安定性については妥協せずにスイッチとネットワークの最大のパフォーマンスを目指しています。Catalyst 4500は、単一のタイムスロットに対して100%でCPUが使用されない限り、CPUは十分には使用されていないと見なします。

Cisco IOS ソフトウェア リリース 12.2(25)EWA2 以降は、CPUのパケット処理とプロセス処理のメカニズムとアカウントを強化しています。そのため、Catalyst 4500の展開にはこれらのリリースを使用してください。

Catalyst 4500 の高 CPU 使用率に関する理由を特定する

Catalyst 4500のCPUパケット処理のアーキテクチャと設計を理解したので、Catalyst 4500のCPU使用率が高い理由を特定できます。Catalyst 4500には、CPUの使用率が高くなる根本原因を特定するために必要なコマンドとツールが用意されています。理由を特定した後、管理者は次のいずれかの操作を実行できます。

- 修正措置：これには、設定やネットワークの変更、またはさらに詳しい分析を依頼する

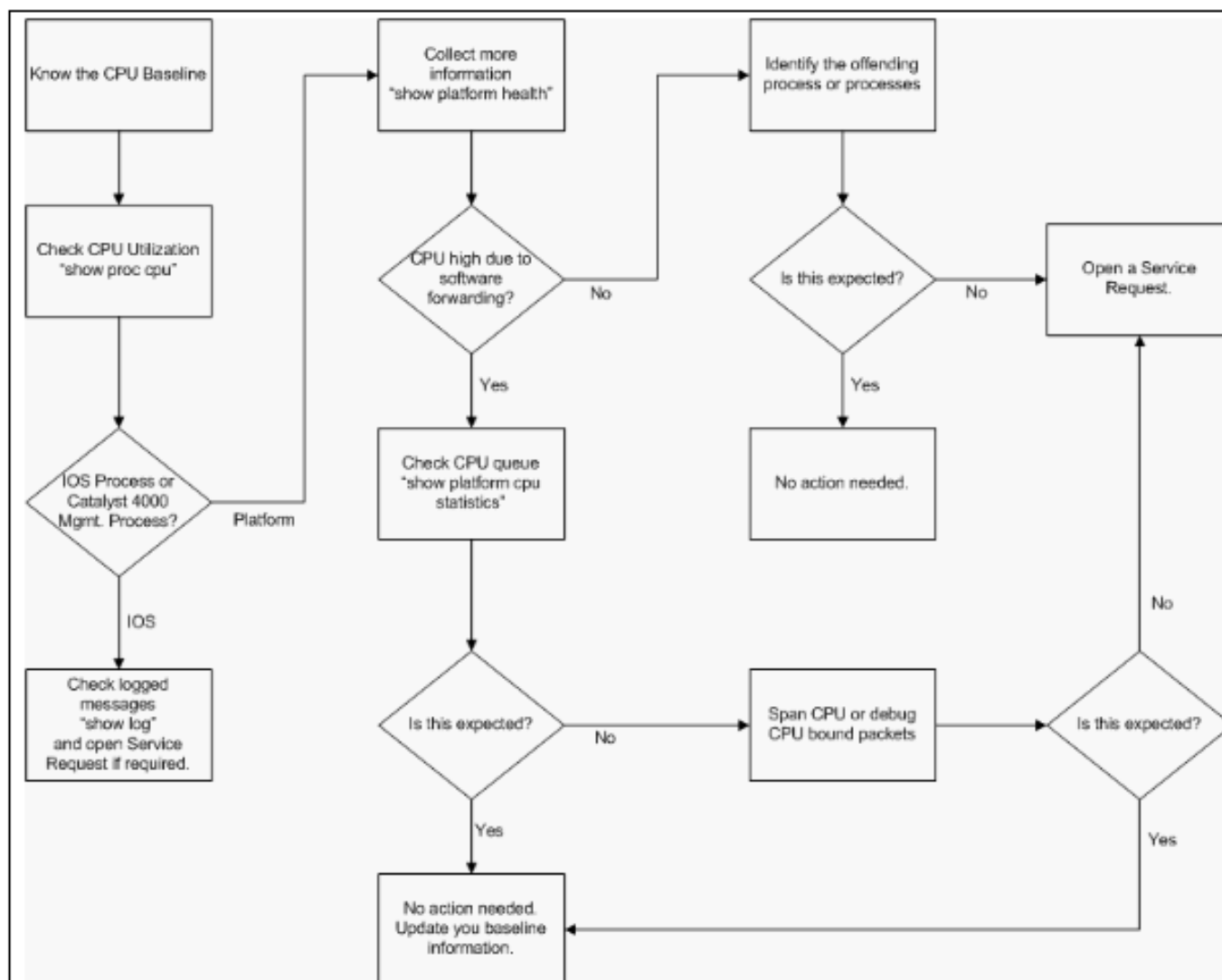
[Ciscoテクニカル](#) サポートサービスリクエストの作成が含まれます。

- 操作なし：Catalyst 4500 は、予想に従って動作します。必要なソフトウェアパケットの転送やバックグラウンド ジョブをすべて実行するためにスーパーバイザ エンジンによって CPU サイクルが最大化されているので、CPU の使用率が高くなっています。

すべての場合で修正操作が必要ない場合であっても、高 CPU 使用率の理由を特定するようにしてください。高 CPU 使用率は、ネットワーク内のある問題の現象に過ぎない可能性があります。その問題の根本原因を解決することが、CPU 使用率を低くするために必要になる可能性があります。

[図 2 は、Catalyst 4500 の高 CPU 使用率の根本原因を特定するために使用するトラブルシューティング方法を示しています。](#)

図 2：Catalyst 4500 スイッチにおける高 CPU 使用率のトラブルシューティング方法



Catalyst 4500スイッチでのCPU高使用率のトラブルシューティング方法

一般的なトラブルシューティング手順は、次のようになります。

1. CPU サイクルを消費している Cisco IOS プロセスを特定するために、show processes cpu コマンドを発行します。
2. プラットフォーム固有のプロセスをさらに特定するために、show platform healthcommand コマンドを発行します。
3. かなりアクティブなプロセスが `K2CpuMan Review` の場合、CPU をヒットするトラフィックのタイプを特定するために show platform cpu packet 統計コマンドを発行します。

このアクティビティが `K2CpuMan Reviewprocess` のせいではない場合、ステップ4を省略してステップ5に進みます。

4. 必要に応じて、「[CPU宛のトラフィックを分析するためのツールのトラブルシューティング](#)」を使用してCPUをヒットしたパケットを特定します。

使用するトラブルシューティング ツールの例としては、CPU Switched Port Analyzer (SPAN; スイッチド ポート アナライザ) があります。

5. 一般的な原因については、このドキュメントの「[一般的な高CPU使用率の問題のトラブルシューティング](#)」セクションを参照してください。

それでも根本原因を特定できない場合は、[シスコテクニカルサポート](#)にお問い合わせください。

基準 CPU 使用状況

最初の重要なステップは、設定とネットワーク セットアップに対するスイッチの CPU 使用状況を知ることです。Catalyst 4500スイッチのCPU使用率を特定するには、show processes cpucommandを使用します。ネットワーク設定に設定を追加したり、ネットワークトラフィックのパターンが変化したりすると、CPU使用率のベースラインを継続的に更新する必要があることがあります。[図2](#)は、この要件を示しています。

次の出力は、完全にロードされた Catalyst 4507R のものです。定常状態の CPU 使用率は、およそ 32 ~ 38 % です。これは、このスイッチで管理機能を実行するために必要とされる比率です。

```
<#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	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

5 秒間の CPU 使用率は次のように表記されます。

x%/y%

x%はCPUの総使用率を表し、andy%は割り込みレベルで消費されたCPUを表します。Catalyst 4500 スイッチのトラブルシューティングを行うときには、合計 CPU 使用率だけに注目してください。

Catalyst 4500 スイッチでの show processes cpu コマンドについて

このshow processes cpuoutputは、CPUを使用する2つのプロセスCat4k Mgmt HiPriandCat4k Mgmt LoPriが存在することを示しています。これらの2つのプロセスが、Catalyst 4500 の基本的な管理機能を実行する複数のプラットフォーム固有プロセスを集約します。これらのプロセスは、コントロールプレーンを処理し、また、ソフトウェアでスイッチングされたり、処理されたりする必要のあるデータ パケットを処理します。

Cat4k Mgmt HiPriandCat4k Mgmt LoPriのコンテキストの下でプラットフォーム固有のプロセスのどれがCPUを使用するのかを確認するには、show platform healthcommandを発行します。

プラットフォーム固有のプロセスそれぞれには、CPU の目標となるまたは予想される使用率があります。CPU 使用率が目標以内であるプロセスは、高い優先順位で実行されます。show processes cpucommand出力は、Cat4k Mgmt HiPriの下での使用率をカウントしています。プロセスが目標/予想使用率を超える場合、そのプロセスは低優先度のコンテキストの下で実行されます。show processes cpucommand出力は、Cat4k Mgmt LoPriの下で追加の使用率をカウントします。ThisCat4k Mgmt LoPriisは、また、一貫性チェックや読み取りインターフェイスカウンタなどのバックグラウンドや他の優先度の低いプロセスの実行にも使用されます。このメカニズムによって、CPU は優先順位の高いプロセスを必要なときに実行し、残されたアイドル状態の CPU サイクルで優先順位の低いプロセスを処理しています。目標の CPU 使用率をわずかに超えたり、使用率が一時的に高くなることは、検査を必要とする問題を示すものではありません。

<#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									
0.02										
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
EthHoTeLinecardMan(6 1.66
0.17
10 6 100 500 0
0 0
6:38
-----
%CPU Totals 212.80
35.63

```

Catalyst 4500 スイッチでの show platform health コマンドについて

show platform healthcommandコマンドは、開発エンジニアだけに関係のある多くの情報を提供します。高いCPU使用率のトラブルシューティングを行うには、出力の%CPU実際の列で高い数値を探します。また、その行の右側を必ず見て、1分と1時間の列%CPU列でそのプロセスのCPU使用率を確認してください。場合によっては、プロセスが一時的にピークに達するものの、CPUを長時間占有することはない場合があります。ハードウェアのプログラミングやそのプログラミングの最適化の際には、一時的にCPUの使用率が高くなることがあります。たとえば、CPU使用率の急上昇は、TCAMの大きなACLのハードウェアプログラミング中は正常なことです。

「Catalyst 4500スイッチでのshow processes cpuコマンドについて」セクションのshow platform healthcommandコマンドの出力では、Stub-JobEventSchedulerとK2CpuMan Reviewprocessesが多くのCPUサイクルを使用しています。表2には、show platform healthcommandの出力に示される、プラットフォーム固有の一般的なプロセスに関する基本情報があります。

表 2 : show platform health コマンドからのプラットフォーム固有プロセスの説明

プラットフォーム固有のプロセスの名前	説明
Pim-review	シャーシ/ラインカードの状態管理
Ebm	エージングや監視などのイーサネットブリッジモジュール
Ac1-Flattener/K2Ac1Man	ACL マージ プロセス
KxAc1PathMan:PathTagMan-Review	ACL 状態管理とメンテナンス
K2CpuMan Review	ソフトウェアパケット転送を実行するプロセス。このプロセスが原因で高いCPU使用率が発生する場合、show platform cpu packet statisticsコマンドの使用によってCPUをヒットするパケットを調査します。
K2AccelPacketMan	CPU から宛先を指定されてパケットを送信するために、パケットエンジンとやり取りするドライバ
K2Ac1CamMan	QoS とセキュリティ機能のための入出力 TCAM ハードウェアを管理します
K2Ac1PolicerTableMan	入出力ポリサーを管理します

K2L2	Catalyst 4500 Cisco IOSソフトウェアのL2転送サブシステムを表します。これらのプロセスは、さまざまなL2テーブルのメンテナンスを担当します。
K2PortMan Review	さまざまなポート関連のプログラミング機能を管理します
K2Fib	FIB ¹ 管理
K2FibFlowCache	PBR ² キャッシュ管理
K2FibAdjMan	FIB 隣接関係テーブル管理
K2FibMulticast	マルチキャスト FIB エントリを管理します
K2MetStatsMan Review	MET ³ 統計情報を管理します
K2QosDb1Man Review	QoS DBL4 を管理します
IrmFibThrottler Thro	IP ルーティング モジュール
K2 L2エージングテーブルRe	L2エージング機能を管理します
GalChassisVp-review	シャーシ状態監視
S2w - ジョブイベントスケジュール	ラインカードの状態を監視するためにS2W ⁵ プロトコルを管理します
Stub-JobEventSchedul	スタブ ASIC ベースのラインカードの監視とメンテナンス
RkiosPortMan Port Re	ポートの状態の監視とメンテナンス
Rkios Module State R	ラインカードの監視とメンテナンス
EthHoleLinecardMan	各ラインカードでGBIC ⁶ を管理します

¹FIB =転送情報ベース

²PBR = policy-based routing (ポリシーベースルーティング) 。

³MET =マルチキャスト拡張テーブル。

⁴DBL =ダイナミックバッファ制限

⁵S2W =シリアル/ワイヤ

⁶GBIC = Gigabit Interface Converter (ギガビットインターフェイスコンバータ)

一般的な CPU 高使用率の問題に関するトラブルシューティング

このセクションでは、Catalyst 4500 スイッチで CPU 使用率が高くなる一般的な問題について説明します。

プロセス交換パケットによる高 CPU 使用率

CPU 使用率が高くなる一般的な原因の 1 つは、ソフトウェアによって転送されるパケットやコントロールパケットの処理によって、Catalyst 4500 の CPU の負荷が高くなることです。ソフトウェアで転送されるパケットの例としては、IPX や BPDU などのコントロールパケットがあります。これらのパケットの数が少ない場合は、通常、CPU に送信されます。ただし、継続的にパケッ

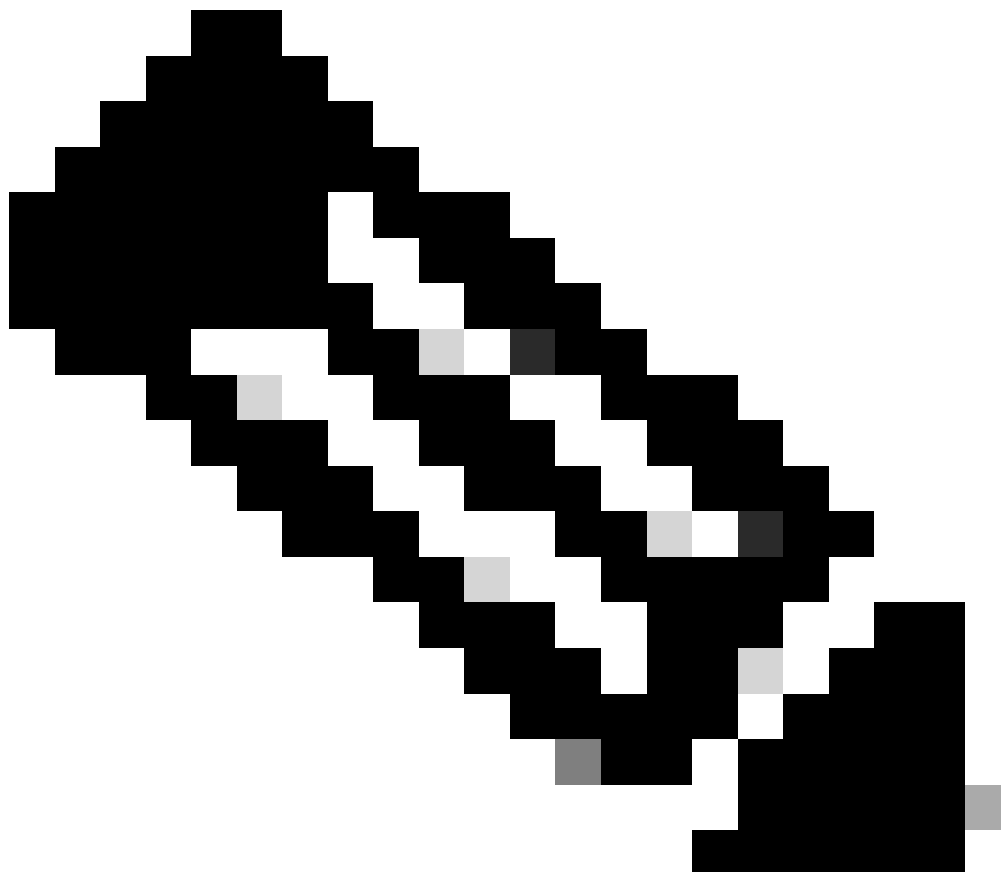
トが多くなると、設定エラーまたはネットワーク イベントがあることを示す可能性があります。処理のためにパケットが CPU に転送されるようになるイベントの原因を特定する必要があります。この特定によって、高 CPU 使用率の問題をデバッグできるようになります。

プロセス交換パケットによって CPU 使用率が高くなる一般的な原因には、次のようなものが挙げられます。

- [スパニング ツリー ポート インスタンスの数が多](#)
- [ICMPリダイレクト、同じインターフェイス上のルーティングパケット](#)
- [IPX または AppleTalk ルーティング](#)
- [ホスト学習](#)
- [セキュリティ ACL 用のハードウェア リソース \(TCAM \) の枯渇](#)
- [ACL内のキーワード](#)
- [レイヤ 2 転送ループ](#)

CPU へのパケット交換のその他の理由には次のものがあります。

- MTU フラグメント化 : パケットのパスにあるすべてのインターフェイスが同一の MTU を持っていることを確認します。
- TCPフラグが established 以外である ACL
- IP バージョン 6 (IPv6) ルーティング : これは、ソフトウェア交換パス経由でだけサポートされます。
- GRE : これは、ソフトウェア交換パス経由でだけサポートされます。
- 入力または出力の Router ACL (RACL; Router ACL) のトラフィック拒否



注：これは、Cisco IOSソフトウェアリリース12.1(13)EW1以降ではレート制限されています。

ACLのインターフェイスの下で`no ip unreachable command`コマンドを発行します。

- 過度のARPとDHCPトラフィックが、直接接続されたホストの数が多いために、処理用にCPUをヒットします。

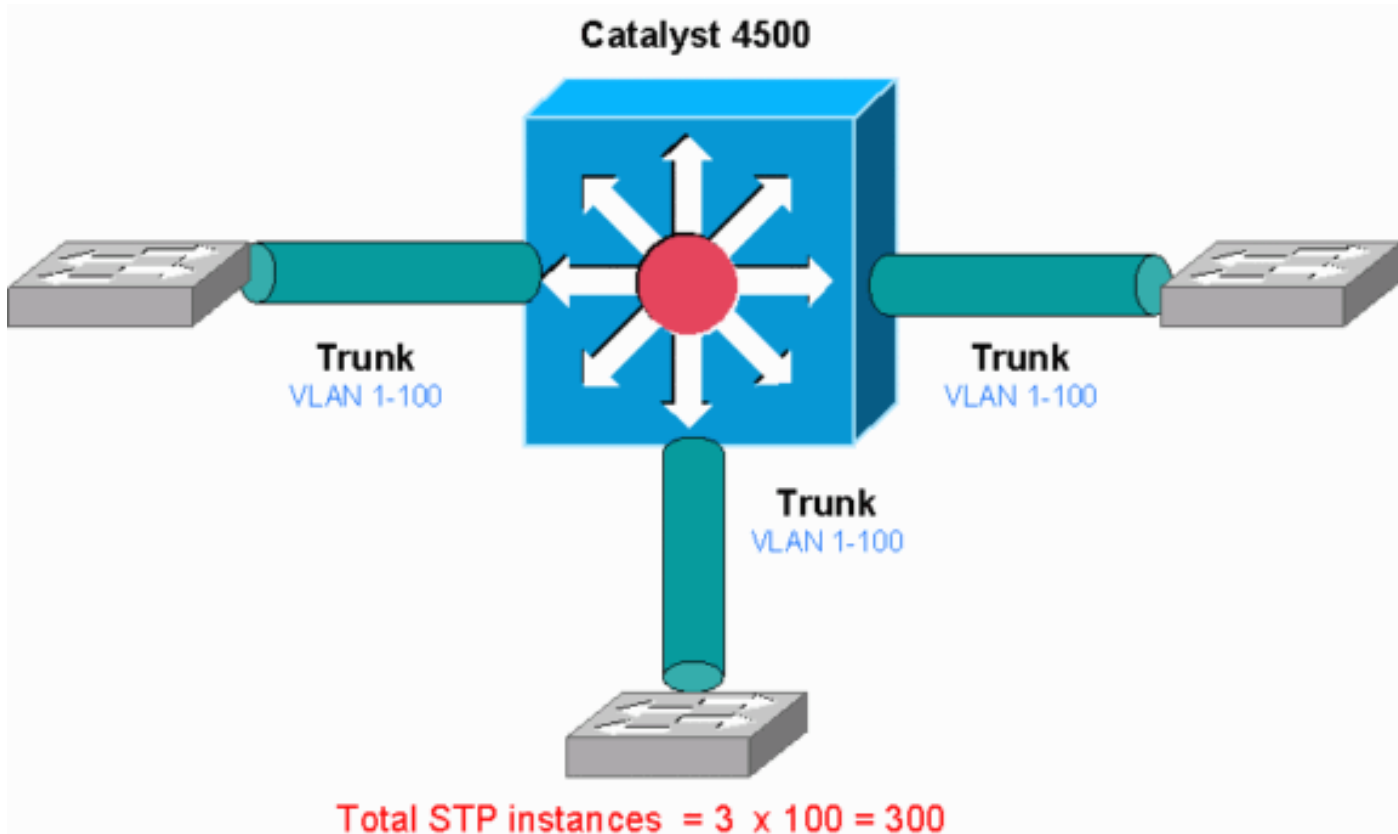
DHCP攻撃が予想される場合は、特定のホストポートからレート制限のあるDHCPトラフィックに対してDHCPスヌーピングを使用します。

- 正規のまたは動作不良のエンドステーションによる過剰なSNMPポーリング

スパニングツリーポートインスタンスの数が多い

Catalyst 4500では、Per VLAN Spanning Tree+ (PVST+; VLAN単位スパニングツリー)モードで、3000個のスパニングツリーポートのインスタンスまたはアクティブポートがサポートされています。これは、Supervisor Engine II+ および II+TS と、Catalyst 4948を除く、すべてのスーパーバイザエンジンでサポートされています。Supervisor Engine II+ および II+TS と、Catalyst

4948 は、最大 1500 のポート インスタンスをサポートしています。これらの STP インスタンスの推奨事項よりも上回る場合、スイッチは高い CPU 使用率を示します。



Catalyst 4500

この図は、それぞれが VLAN 1 ~ 100 を伝送する 3 つのトランク ポート を備えた Catalyst 4500 を示しています。これは 300 のスパニング ツリー ポート インスタンスと同じです。一般に、スパニング ツリー ポート インスタンスは次の式で計算できます。

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

図ではアクセス ポートはありませんが、3 つのトランクは VLAN を 1 から 100 まで伝送します。

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

ステップ1 : コマンドを使用して、Cisco IOSプロセスを確認 `show processes cpu` します。

このセクションでは、CPU 使用率が高くなる原因を絞り込むために使用するコマンドについて説明します。 `show processes cpu` コマンドを発行すると、2つの主なプロセス、 `Cat4k Mgmt LoPrimarySpanning Tree` が主にCPUを使用していることがわかります。この情報だけで、スパニングツリー プロセスによって CPU サイクルの大部分が消費されていることがわかります。

<#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	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

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

ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。

プラットフォーム固有のどのプロセスがCPUを消費しているのかを理解するには、`show platform healthcommand`コマンドを発行します。この出力から、CPUに送られるパケットを処理するジョブである**K2CpuMan Reviewprocess**がCPUを使い切っていることがわかります。

```
<#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
```

ステップ3:CPUに送られるトラフィックのタイプを特定するために、トラフィックを受信するCPUキューを確認します。

どのCPUキューがCPUに送られるパケットを受信するのかを調べるために、`show platform cpu packet statistics`コマンドを発行します。このセクションの出力を見ると、コントロール キューが大量のパケットを受信していることがわかります。[表1](#)の情報と、[ステップ1](#)で調べた結果を使用します。CPU が処理しているパケットと、CPU 使用率が高くなっている原因が、BPDU 処理であることがわかります。

```
<#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

手順4：根本原因を特定します。

show spanning-tree summaryコマンドを発行します。スパンニングツリー ポートに大量のインスタンスが存在しているために BPDUを受信しているかどうかチェックできます。出力が根本原因を明確に特定します。

<#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 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

PVST+ モード設定の VLAN が多数存在しています。問題を解決するには、STP モードを Multiple Spanning Tree (MST; 多重スパンニング ツリー) に変更します。ときには、数多くの VLAN がすべてのトランク ポートに転送されるために、STP インスタンスの数が多くなることがあります。この場合、STP アクティブポートの数を推奨値より低く抑えるために、不要な VLAN をトランクから手動でブルーニングします。

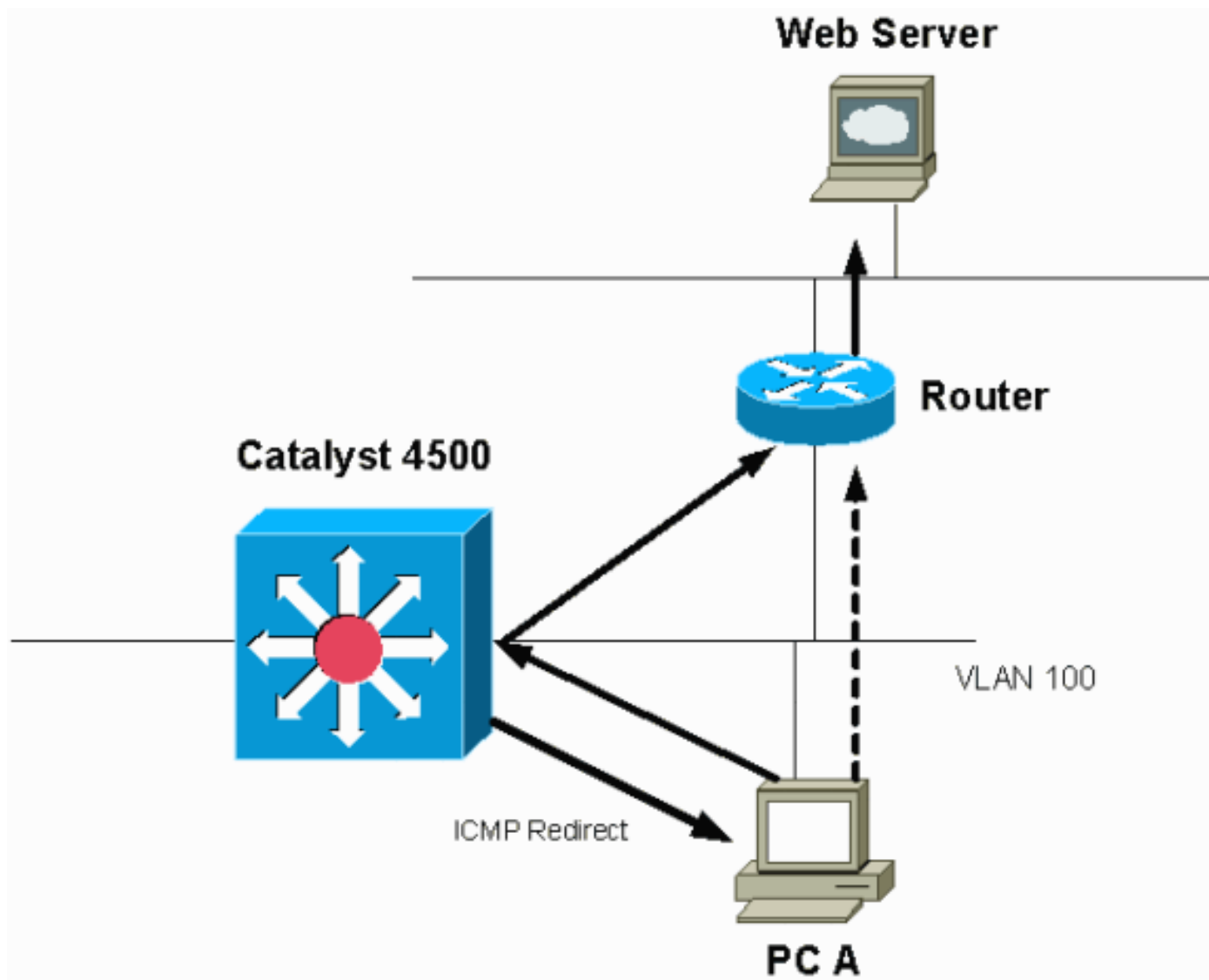
ヒント：IP Phone ポートをトランクポートとして設定していないことを確認します。これはよく起こる設定ミスです。IP Phone ポートは音声 VLAN 設定で設定します。この設定は、擬似トランクを作成しますが、不必要な VLAN を手動でブルーニングする必要はありません。音声ポートの設定方法の詳細については、『[音声インターフェイスの設定](#)』ソフトウェア設定ガイドを参照してください。Cisco 以外の IP Phone は、この音声 VLAN や補助 VLAN 設定をサポートしません。シスコ以外の IP phone を接続しているポートは、手作業で削除する必要があります。

ICMP リダイレクト、同じインターフェイス上のルーティングパケット

同じインターフェイス上のルーティングパケット、または同じ L3 インターフェイス上の入力および出カトラフィックは、スイッチによる ICMP リダイレクトの原因となる可能性があります。最終的な宛先へのネクスト ホップ デバイスが送信元のデバイスと同じサブネット内にあるのをスイッチが知っている場合、スイッチは発信元への ICMP リダイレクトを生成します。このリダイレクトメッセージは、ネクスト ホップ デバイスにパケットを直接送信するよう送信元に指示するものです。このメッセージは、宛先に対する経路として、ネクスト ホップ デバイスの方がより適切である (このスイッチよりもホップが 1 つ少ない) ことを示しています。

このセクションの図において、PC A は Web サーバと通信しています。PC A のデフォルト ゲートウェイは、VLAN 100 のインターフェイスの IP アドレスを指しています。ただし、Catalyst 4500 の宛先への到達をイネーブルにするネクスト ホップ ルータは、PC A と同じサブネット内にあります。この場合の最良のパスは、「Router」に直接送信することです。Catalyst 4500 は ICMP リダ

イレクトメッセージを PC A に送信します。このメッセージは、PC A に対して、Catalyst 4500 を経由するのではなく、Router を経由して Web サーバを宛先とするパケットを送信するように指示しています。ただし、ほとんどの場合、エンドデバイスは ICMP リダイレクトには応答しません。応答がないことによって、Catalyst 4500 は入力パケットと同じインターフェイス経由で転送するすべてのパケットに対してこれらの ICMP リダイレクトの生成を行うので、これが Catalyst 4500 が多くの CPU サイクルを消費する原因となります。



デフォルトでは *ICMP* リダイレクトはイネーブルになっています。

デフォルトでは *ICMP* リダイレクトはイネーブルになっています。ディセーブルにするには、`ip icmp redirects` コマンドを使用します。対応する SVI または L3 のインターフェイスでこのコマンドを発行します。



注: `ip icmp redirects` はデフォルトのコマンドであるため、`show running-configuration` コマンドの出力には表示されません。

ステップ1: `show processes cpu` コマンドでCisco IOSプロセスを確認します。

`show processes cpu` コマンドを発行します。2つの主なプロセスである `Cat4k Mgmt LoPrimaryIP Input` が主にCPUを使用していることがわかります。この情報だけで、このIPパケットの処理によってCPUの大部分が消費されていることがわかります。

<#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
----	-----------	-----------	-----	--------	--------	--------	---	----------

ステップ2: show platform health コマンドでCatalyst 4500固有のプロセスを確認します。

show platform healthコマンドの出力は、CPUに送られるパケットを処理するためにCPUの使用を確認します。

```
<#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

ステップ3:CPUに送られるトラフィックのタイプを特定するために、トラフィックを受信するCPUキューを確認します。

どのCPUキューがCPUに送られるパケットを受信するのかを調べるために、show platform cpu packet statisticsコマンドを発行します。L3 Fwd Lowqueueが大量のトラフィックを受信していることがわかります。

```
<#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	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

手順4：根本原因を特定します。

この場合、CPU をヒットするトラフィックを判断するために CPU SPAN を使用します。CPU SPANの詳細については、このドキュメントの「[ツール1:SPANでCPUトラフィックを監視する：Cisco IOSソフトウェアリリース12.1\(19\)EW以降](#)」のセクションを参照してください。show running-configurationコマンドを使用すると、トラフィックと設定の解析が実行できます。この場合、パケットは同じインターフェイス経由でルーティングされ、これが各パケットの ICMP リダイレクトの発行につながります。これは、Catalyst 4500 で CPU の使用率が高くなる一般的な原因の 1 つです。

発信元デバイスは、Catalyst 4500が送信するICMPリダイレクトに従って動作し、宛先のネクストホップを変更します。しかし、すべてのデバイスが ICMP リダイレクトに回答するとは限りません。デバイスが応答しない場合、Catalyst 4500 はスイッチが送信元デバイスから受け取るすべてのパケット用にリダイレクトを送信する必要があります。これらのリダイレクトは、かなりの量の CPU リソースを消費する可能性があります。この解決策は、ICMP リダイレクトを無効にすることです。インターフェイスでno ip redirectsコマンドを発行します。

このシナリオは、セカンダリ IP アドレスも設定しているときに発生することがあります。セカンダリ IP アドレスを有効にすると、IP リダイレクトは自動的に無効になります。IP リダイレクトは手動でイネーブルにはしないでください。

[thisICMP Redirects: Routing Packets on the Same Interfaces](#) セクションが示すように、ほとんどのエンドデバイスはICMPリダイレクトには応答しません。そのため、一般的には、この機能はディセーブルにします。

IPX または AppleTalk ルーティング

Catalyst 4500 は、ソフトウェアで転送されるパスだけを經由する IPX および AppleTalk ルーティングをサポートします。このようなプロトコルを設定している場合は、CPU の使用率が高くなることは異常ではありません。



注：同じVLANでのIPXトラフィックとAppleTalkトラフィックのスイッチングには、プロセス交換は必要ありません。ルーティングされる必要のあるパケットだけがソフトウェアパス転送を必要とします。

ステップ1: show processes cpu コマンドでCisco IOSプロセスを確認します。

どのCisco IOSプロセスがCPUを消費するかを確認するために、**show processes cpu** コマンドを発行します。このコマンド出力では、トッププロセスが**Cat4k Mgmt LoPri**であることがわかります。

<#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
----	-----------	-----------	-----	--------	--------	--------	---	------------------

ステップ2: show platform health コマンドでCatalyst 4500固有のプロセスを確認します。

show platform healthコマンドの出力は、CPUに送られるパケットを処理するためにCPUの使用を確認します。

<#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:
```

```
K2Acce1PacketMan: Tx 10.00 8.03 20 0 100 500 21 29 26 270:4
```

ステップ3: トラフィックを受信するCPUキューを確認し、CPUに送られるトラフィックのタイプを特定します。

CPUをヒットするトラフィックのタイプを判断するために、show platform cpu packet statisticsコマンドを発行します。

```
<#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
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, unreachable	11013420	9	8	8	8

手順4：根本原因を特定します。

管理者はIPXまたはAppleTalkルーティングを設定しているため、根本原因の特定は簡単である必要があります。しかし、確認するには、CPUトラフィックをSPANして、そのトラフィックが期待していたトラフィックであることを確認する必要があります。CPU SPANの詳細については、このドキュメントの「[ツール1:SPANでCPUトラフィックを監視する : Cisco IOSソフトウェアリリース12.1\(19\)EW以降](#)」のセクションを参照してください。

この場合、管理者は基準CPUを現在の値に更新する必要があります。Catalyst 4500 CPUは、CPUがソフトウェアによるパケット交換を処理している場合には、予想どおりに動作します。

ホスト学習

Catalyst 4500は、MACアドレスがMACアドレステーブル内にまだない場合には、さまざまなホストのMACアドレスを学習します。スイッチングエンジンは、新しいMACアドレスのパケットのコピーをCPUに転送します。

すべてのVLANインターフェイス(レイヤ3)は、シャーシの基本ハードウェアアドレスをそのMACアドレスとして使用します。その結果、MACアドレステーブルにはエントリが存在しないことになり、これらのVLANインターフェイスを宛先とするパケットは処理のためにCPUに送信されることはありません。

スイッチが学習する新しいMACアドレスの数が多すぎる場合、結果として高いCPU使用率になります。

ステップ1: show processes cpu コマンドでCisco IOSプロセスを確認します。

どのCisco IOSプロセス show processes cpuがCPUを消費するかを確認するために、コマンドを発行します。このコマンド出力では、トッププロセスがCat4k Mgmt LoPriであることがわかります。

```
<#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

```

ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。

show platform healthcommandの出力は、CPUに送られるパケットを処理するためにCPUの使用を確認します。

<#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

```

ステップ3：トラフィックを受信するCPUキューを確認し、CPUに送られるトラフィックのタイプを特定します。

CPUをヒットするトラフィックのタイプを判断するために、`show platform cpu packet statistics`コマンドを発行します。

<#root>

Switch#

`show platform cpu packet statistics`

!--- Output suppressed.

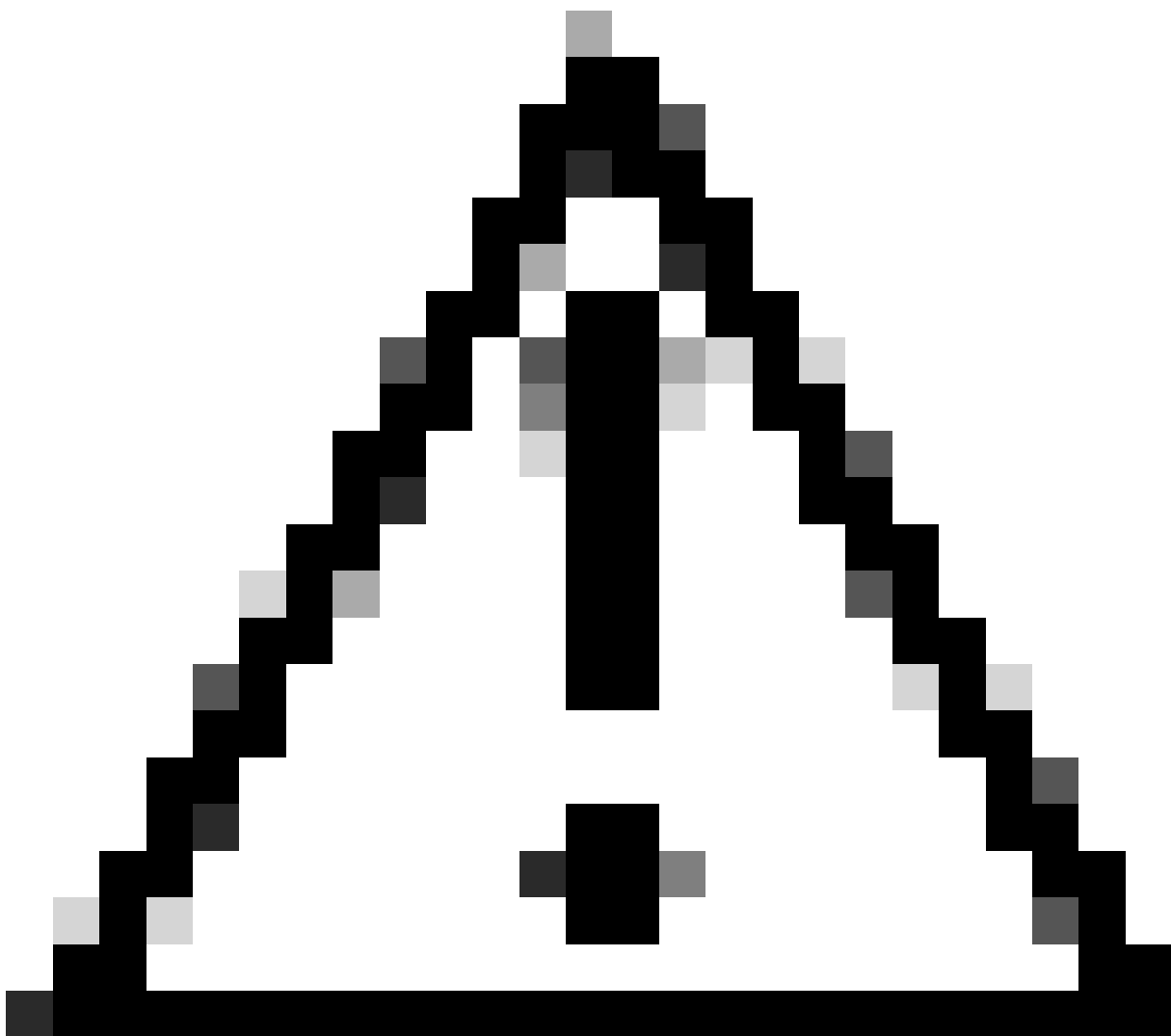
Packets Received by Packet Queue

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, unreach	11013420	9	8	8	8

手順4：根本原因を特定します。

show platform healthコマンドの出力は、CPUが多くの新しいMACアドレスを確認していることを示しています。この状況は多くの場合、ネットワーク トポロジの不安定に起因します。たとえば、スパニング ツリー トポロジが変更した場合、スイッチは Topology Change Notification (TCN; トポロジ変更通知) を生成します。TCN の発行によって、PVST+ モードでのエージング タイムが 15 秒に短縮されます。MAC アドレス エントリは、アドレスがその時間内に学習されない場合には、フラッシュされます。Rapid STP (RSTP) (IEEE 802.1w) または MST (IEEE 802.1s) の場合、TCN が別のスイッチからのものであれば、エントリは即座にエージングアウトされます。このエージングアウトによって、MAC アドレスが新たに学習されます。これは、トポロジの変更があまりない場合には重要な問題にはなりません。しかし、フラップ リンク、障害のあるスイッチ、PortFast 用にイネーブルになっていないホスト ポートがあると、トポロジの変更が過剰に発生する可能性があります。結果として、MAC テーブルのフラッシュやその後の再学習が頻繁に発生します。根本原因を明らかにするための次のステップは、ネットワークのトラブルシューティングです。スイッチは、予想されたとおりに動作し、ホスト アドレス学習のために CPU にパケットを送信します。過剰な TCN になる障害のあるデバイスを特定して修理します。

使用しているネットワークに、バースト状態でトラフィックを送信しているデバイスが多数あり、スイッチ上で MAC アドレスがエージングアウトして、アドレスの再学習を発生させている可能性があります。この場合、いくらかの余裕を提供するために MAC アドレス テーブルのエージング タイムを増やしてください。エージング タイムを長くすると、スイッチがデバイスの MAC アドレスをエージングアウトさせずにテーブル内に保持している時間が長くなります。



注意：このエージングアウト変更は、十分に考慮した上で行ってください。この変更は、ネットワーク内のデバイスがモバイルである場合には、トラフィックをブラックホールに送信してしまう可能性があります。

セキュリティ ACL 用のハードウェア リソース (TCAM) の枯渇

Catalyst 4500 では、Cisco TCAM を使用して、設定された ACL をプログラムしています。TCAM では、ハードウェア転送パスで ACL の適用ができます。転送パスに ACL があってもなくても、スイッチのパフォーマンスには影響がありません。パフォーマンスは、ACL ルックアップのパフォーマンスがライン レートで行われるので、ACL のサイズに関係なく一定です。ただし、TCAM は有限のリソースです。そのため、過剰な数の ACL エントリを設定すると、TCAM の容量を超過します。表3は、Catalyst 4500 スーパーバイザエンジンとスイッチのそれぞれで使用可能な TCAM リソースの数を示しています。

表3:Catalyst 4500 Supervisor Engine/スイッチのTCAM容量

製品	Feature TCAM (方向ごと)	QoS TCAM (方向ごと)
スーパーバイザエンジンII+/III+TS	1024 個のマスクのある 8192 エントリ	1024 個のマスクのある 8192 エントリ
Supervisor Engine III/IV/V および Catalyst 4948	2048 個のマスクのある 16,384 エントリ	2048 個のマスクのある 16,384 エントリ
Supervisor Engine V-10GE および Catalyst 4948-10GE	16,384 個のマスクのある 16,384 エントリ	16,384 個のマスクのある 16,384 エントリ

スイッチでは、Feature TCAM を使用して、RACL や VLAN ACL (VACL) などのセキュリティ ACL をプログラムしています。また、スイッチは、ダイナミック ACL 用の IP Source Guard (IPSG; IP ソース ガード) のようなセキュリティ機能のために、Feature TCAM を使用します。さらにスイッチでは、QoS TCAM を使用して、分類とポリサー ACL をプログラムしています。

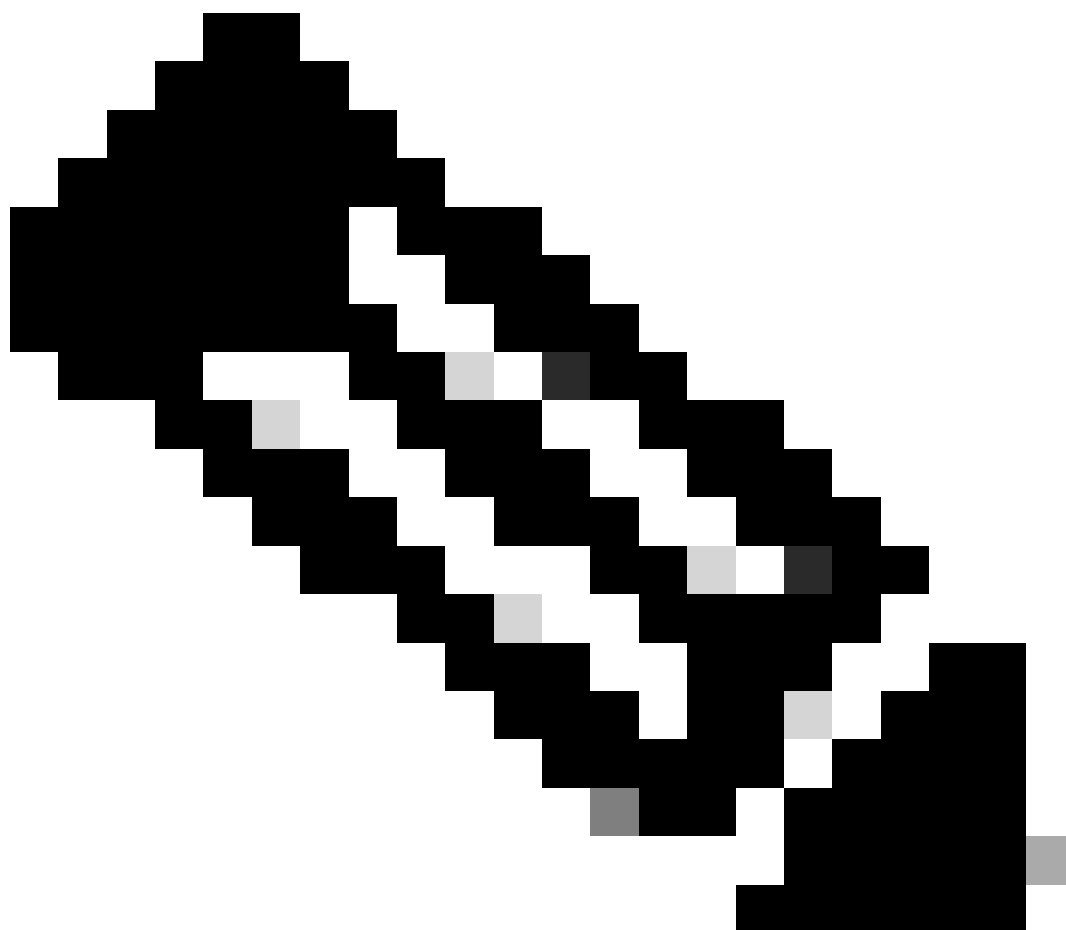
セキュリティ ACL のプログラミング中に Catalyst 4500 の TCAM リソースがなくなった場合、ACL の部分的な適用がソフトウェアパスを経由して発生します。この ACE にヒットしたパケットはソフトウェアで処理され、これが CPU 使用率が高くなる原因になります。ACL はトップダウンでプログラムされます。いいかえると、ACL が TCAM に収まらない場合、ACL の下部にある ACE は TCAM ではプログラムされない傾向にあります。

TCAM のオーバーフローが起きると、次の警告メッセージが表示されます。

```
%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.
```

このエラーメッセージは、show loggingcommandの出力に表示されます。最終的に、このメッセージは、一部のソフトウェア処理

が実行され、その結果として高いCPU使用率が発生する可能性があることを示しています。



注：大きなACLを変更した場合、変更したACLがTCAMに再びプログラムされる前に、このメッセージが短時間表示されることがあります。

ステップ1: `show processes cpu` コマンドを使用して、Cisco IOS プロセスを確認します。

`show processes cpu` コマンドを発行します。Cat4k Mgmt LoPriprocess が CPU サイクルのほとんどを占有するため、CPU 使用率が高いことが確認できます。

<#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
----	-------	---------	----	-------	-------	-------	---	------------------

ステップ2: show platform health コマンドでCatalyst 4500固有のプロセスを確認します。

show platform healthコマンドを発行します。CPUに送られるパケットを処理するジョブである**K2CpuMan Review**がCPUを使用していることがわかります。

<#root>

Switch#

show platform health

%CPU	%CPU	RunTimeMax	Priority		Average %CPU		Total		5Sec	Min	Hour	CPU
			Target	Actual	Target	Actual	Fg	Bg				
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	
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	
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		

ステップ3: トラフィックを受信するCPUキューを確認し、CPUに送られるトラフィックのタイプを特定します。

どのCPUキューであるか、つまりどのタイプのトラフィックがCPUキューをヒットしているかを調べる必要があります。`show platform cpu packet statistics`コマンドを発行します。ACL sw processingqueueが大量のパケットを受信していることがわかります。したがって、TCAMのオーバーフローによって、CPU使用率が高くなっています。

<#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
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

ステップ4：問題を解決します。

[ステップ3](#)では、このシナリオの根本原因を判別しました。オーバーフローの原因となった ACL を削除するか、または、オーバーフローを避けるために ACL を最小化します。また、『[ACLによるネットワークセキュリティの設定](#)』設定ガイドラインを検討して、ACL設定とハードウェアのプログラミングを最適化します。

ACL 内の log キーワード

Catalyst 4500 では、特定の ACL エントリにヒットしたパケットの詳細をロギングすることをサポートしていますが、ロギングを過剰に行うと CPU の使用率が高くなる場合があります。トラフィック検出段階を除き、logkeywordsの使用は避けてください。トラフィック検出段階では、明示的に ACE を設定していないネットワークを経由して伝送されるトラフィックを特定します。統計情報を収集するためにlogkeywordを使用しないでください。Cisco IOSソフトウェアリリース12.1(13)EW以降では、メッセージはレート制限されています。ACLに一致するパケットの数をカウントするためにログメッセージを使用する場合、そのカウントは正確ではありません。その代わりに、正確な統計情報を得るためにshow access-listコマンドを使用します。この根本原因の特定は、設定またはロギングメッセージのレビューがACLロギング機能の使用を示す可能性があるため、簡単です。

ステップ1:show processes cpuコマンドを使用して、Cisco IOSプロセスを確認します。

どのCisco IOSプロセスがCPUを消費するかを確認するために、show processes cpuを発行します。このコマンド出力では、トッププロセスがCat4k Mgmt LoPriであることがわかります。

<#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
----	-------	---------	----	-------	-------	-------	---	------------------

ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。

CPUを使用するプラットフォーム固有のプロセスを確認します。show platform healthコマンドを発行します。出力では、**K2CpuMan Reviewprocess**がほとんどのCPUサイクルを使用していることに注意してください。このアクティビティは、CPUがCPU宛てに送られたパケットの処理で負荷が高くなっていることを示しています。

<#root>

Switch#

show platform health

	%CPU		RunTimeMax		Priority		Average			Total CPU
	Target	Actual	Target	Actual	Fg	Bg	5Sec	Min	Hour	
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

ステップ3: トラフィックを受信するCPUキューを確認し、CPUに送られるトラフィックのタイプを特定します。

CPUをヒットするトラフィックのタイプを判断するために、コマンドを発行 **show platform cpu packet statistics**します。このコマンド出力では、パケットの受信が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

ステップ4:問題を解決します。

[ステップ3](#)では、このシナリオの根本原因を判別しました。この問題を回避するには、ACLからogkeywordを削除します。Cisco IOS ソフトウェア リリース 12.1(13)EW1 以降では、CPU 使用率が高くなりすぎないように、パケットはレート制限されています。ACL ヒットを追跡する方法として、アクセス リスト カウンタを使用します。コマンド出力でアクセスリストカウンタを確認 `show access-list acl_id`できます。

レイヤ 2 転送ループ

レイヤ 2 転送ループは、Spanning Tree Protocol (STP; スパニング ツリー プロトコル) の貧弱な実装と、STP に影響する可能性のあるさまざまな問題によって引き起こされる可能性があります。

ステップ1:show processes cpu コマンドでCisco IOSプロセスを確認します

このセクションでは、CPU 使用率が高くなる原因を絞り込むために使用するコマンドについて説明します。show processes cpu コマンドを発行すると、2つの主なプロセス、Cat4k Mgmt LoPrimarySpanning Treeが主にCPUを使用していることがわかります。この情報だけで、スパニングツリー プロセスによって CPU サイクルの大部分が消費されていることがわかります。

<#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

ステップ2:show platform healthコマンドでCatalyst 4500固有のプロセスを確認します

プラットフォーム固有のどのプロセスがCPUを消費しているのかを理解するには、show platform healthコマンドを発行します。この出力から、CPUに送られるパケットを処理するジョブであるK2CpuMan ReviewprocessがCPUを使い切っていることがわかります。

<#root>

Switch#

show platform health

%CPU	%CPU	RunTimeMax	Priority	Average	%CPU	Total					
		Target	Actual	Target	Actual	Fg	Bg	5Sec	Min	Hour	CPU
<i>!--- Output suppressed.</i>											
TagMan-RecreateMtegR	1.00	0.00	10	0	100	500	0	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	0:36
K2Acce1PacketMan: Au	0.10	0.00	0	0	100	500	0	0	0	0	0:00
K2Ac1Man-taggedFlatA	1.00	0.00	10	0	100	500	0	0	0	0	0:00

ステップ3:CPUに送られるトラフィックのタイプを特定するために、トラフィックを受信するCPUキューを確認します

どのCPUキューがCPUに送られるパケットを受信するのかを調べるために、show platform cpu packet statisticsコマンドを発行します。このセクションの出力を見ると、コントロール キューが大量のパケットを受信していることが分かります。[表1](#)の情報と、[ステップ1](#)で調べた結果を使用します。CPU が処理しているパケットと、CPU 使用率が高くなっている原因が、BPDU 処理であることが分かります。

<#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
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

ステップ4：根本原因を特定し、問題を修正する

一般に、トラブルシューティングのために次のステップを実行できます（状況によっては、いくつかのステップは必要ありません）。

-

ループを識別します。

-

ループの範囲を検出します。

•

ループを遮断します。

•

ループの原因を修正します。

•

冗長性を復元します。

各ステップの詳細については、「[転送ループのトラブルシューティング：Cisco IOSシステムソフトウェアが稼働するCatalystスイッチのSTPのトラブルシューティング](#)」を参照してください。

ステップ5：高度なSTP機能の実装

•

BDPU ガード：PortFast をイネーブルにしたポートに接続された未承認のネットワーク デバイスからの STP のセキュリティを保護します。詳細は、「[スパニングツリーPortFast BPDUガードの機能拡張](#)」を参照してください。

•

ループ ガード：レイヤ 2 ネットワークの安定性を増加させます。詳細は、「[ループガードとBPDUスキュー検出機能によるスパニングツリープロトコルの拡張機能](#)」を参照してください。

•

ルート ガード：ネットワーク内のルート ブリッジの配置を確保します。詳細は、「[スパニングツリープロトコルルートガードの強化](#)」を参照してください。

•

UDLD：単方向リンクを検出し、転送ループを防止します。詳細は、「[単方向リンク検出プロトコル機能の説明と設定](#)」を参照してください。

CPU の使用率が高くなるその他の原因

CPU 使用率が高くなるその他の原因としては、次のものがあります。

•

[過剰なリンクフラップ](#)

•

[FIBの一貫性チェックによってCPU使用率が一時的に高くなる](#)

•

[K2FibAdjMan Host Moveprocessでの高いCPU使用率](#)

•

[RkiosPortManポートの確認プロセスでの高いCPU使用率](#)

•

[トランクポートの使用でIP Phoneに接続されたときの高いCPU使用率](#)

•

[RSPANとレイヤ3コントロールパケットを伴う高いCPU使用率](#)

•

大きなACLのプログラミング中の急上昇

CPU使用率の急上昇は、あるインターフェイスから大きなACLの適用や削除を行っている間に発生します。

過剰なリンクフラップ

Catalyst 4500では、割り当てられている1つまたは複数のリンクでフラップが過剰に発生し始めると、CPU使用率が高くなります。この状況は、Cisco IOS ソフトウェア リリース 12.2(20)EWA より前の Cisco IOS ソフトウェア リリースで発生します。

ステップ1: show processes cpu コマンドを使用して、Cisco IOS プロセスを確認します。

どのCisco IOS プロセスがCPUを消費するかを確認するために、show processes cpu command を発行します。このコマンド出力では、トッププロセスが Cat4k Mgmt LoPri であることがわかります。

<#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
----	------	---------	---	-------	-------	-------	---	------------------

ステップ2:show platform healthコマンドを使用して、Catalyst 4500固有のプロセスを確認します。

theshow platform healthcommandの出力は、KxAc1PathMan createprocessがCPUを使い切っていることを示します。このプロセスは、内部パスを作成するためのものです。

<#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	0.03	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
KxAc1PathMan create/	1.00	69.11	10	5	100	500	42	53	22	715:0
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

手順3：根本原因を特定します。

リンク アップ/ダウンのメッセージ用にロギングをイネーブルにします。このロギングはデフォルトで有効になっています。このイネーブル化によって、問題のリンクを非常に短時間で絞り込むことができます。すべてのインターフェイスの下で**logging event link-statuscommand**コマンドを発行します。次の例に示すように、ある範囲内のインターフェイスを便利にイネーブルにするために、**interface** コマンドを使用できます。

```
<#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
```

故障のあるまたはフラッピングしているインターフェイスを特定した後、高い CPU 使用率の問題を解決するために、インターフ

エイスをシャットダウンします。Cisco IOS ソフトウェア リリース 12.2(20)EWA 以降では、このフラッピング リンク状態に対する Catalyst 4500 の動作を向上させています。したがって、改善前ほどには CPU への影響は大きくありません。このプロセスはバックグラウンド プロセスであることに注意してください。この問題による高い CPU 使用率は、Catalyst 4500 スイッチへの悪影響を引き起こしません。

FIB の一貫性チェックによって CPU 使用率が一時的に高くなる

Catalyst 4500 は、FIB テーブルの一貫性チェック中に、CPU 使用率の一時的な急上昇を示すことがあります。FIB テーブルは、CEF プロセスが作成する L3 転送テーブルです。一貫性チェックは、Cisco IOS ソフトウェア FIB テーブルとハードウェア エントリの間の一貫性をチェックします。この整合性により、パケットのルーティングの誤りが生じないようにしています。チェックは、2 秒ごとに発生し、低優先度のバックグラウンド プロセスとして動作します。このプロセスは、通常の動作であり、他の高優先度のプロセスまたはパケットには干渉しません。

show platform healthcommandの出力は、K2Fib ConsistencyがCPUのほとんどを消費していることを示しています。

注：このプロセスの平均CPU使用率は1分または1時間で有意ではなく、これはチェックが短時間の定期的なレビューであることを裏付けます。このバックグラウンド プロセスは、アイドル状態の CPU サイクルしか使用しません。

<#root>

Switch#

show platform health

	%CPU		RunTimeMax		Priority		Average %CPU			Total CPU
	Target	Actual	Target	Actual	Fg	Bg	5Sec	Min	Hour	
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.

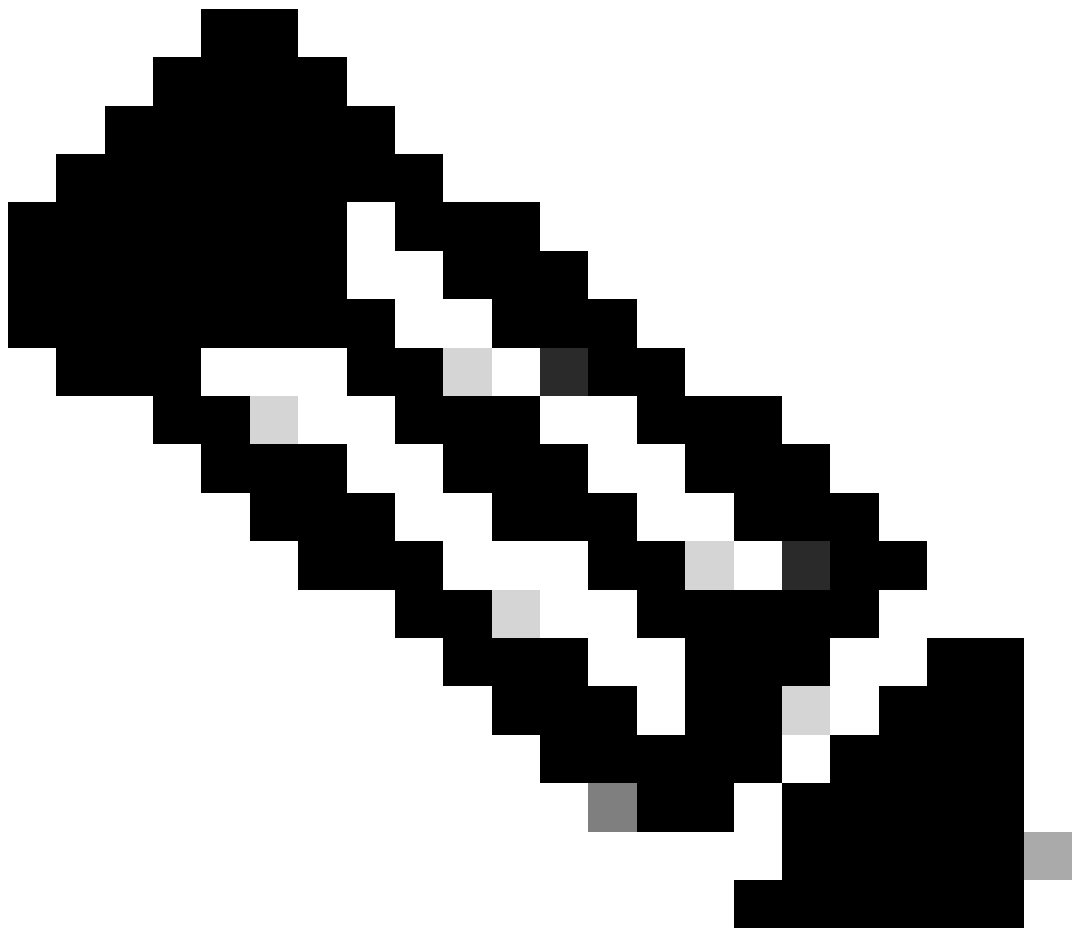
K2Fib cam usage revi	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib IrmFib Review	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib Vrf Default Ro	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib AdjRepop Revie	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib Vrf Unpunt Rev	2.00	0.01	15	0	100	500	0	0	0	0:23
K2Fib Consistency Ch	1.00	60.40	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 Host Move プロセスでの高い CPU 使用率

Catalyst 4500では、**K2FibAdjMan Host Move**processで高いCPU使用率が表示される場合があります。この高い使用率は**show platform healthcommand**コマンドの出力に表示されます。多くの MAC アドレスが新しいポートで頻繁に期限切れになったり、学習されたりすると、この高い CPU 使用率を引き起こします。mac-address-table aging-time のデフォルト値は 5 分 (300 秒) です。この問題の回避策は、MAC アドレス エージング タイムを増やすか、MAC アドレス移動の高い数値を避けるためにネットワークを変更することです。Cisco IOS ソフトウェア リリース 12.2(18)EW 以降は、CPU の消費を少なくするためにこのプロセスの動作を強化しています。Cisco Bug [IDCSCed15021](#)を参照してください。



注：シスコの内部ツールおよび情報にアクセスできるのは、登録ユーザのみです。

<#root>

Switch#

show platform health

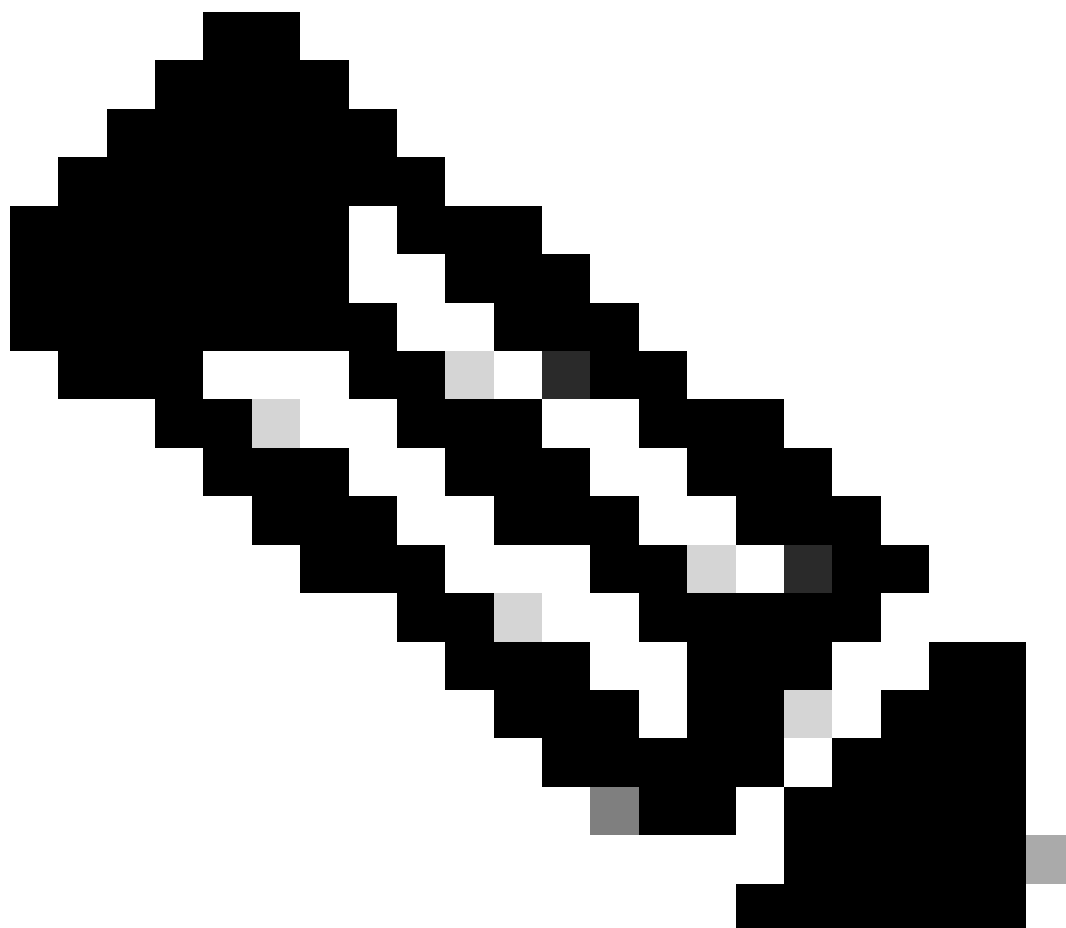
	%CPU Target	%CPU Actual	RunTimeMax Target	RunTimeMax Actual	Priority Fg	Priority Bg	Average 5Sec	%CPU Min	%CPU Hour	Total CPU
Lj-poll	1.00	0.02	2	1	100	500	0	0	0	1:09
GalChassisVp-review	3.00	0.29	10	3	100	500	0	0	0	11:15
S2w-JobEventSchedule	10.00	0.32	10	7	100	500	0	0	0	10:14
<i>!--- Output suppressed.</i>										
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

グローバル設定モードでは、MACアドレスの最大エイジング タイムを変更できます。コマンド構文はルータの場合は `ismac-address-table aging-time seconds`、Catalystスイッチの場合は `mac-address-table aging-time seconds [vlan vlan-id]` です。詳細は、『[Cisco IOSスイッチングサービスコマンドリファレンスガイド](#)』を参照してください。

RkiosPortMan Port Review プロセスにおける高い CPU 使用率

Catalyst

4500は、Cisco IOSソフトウェアリリース12.2(25)EWAおよび12.2(25)EWA1の**theshow platform healthcommand**の出力で**RkiosPortMan Port Reviewprocess**の高いCPU使用率を表示することがあります。Ciscoバグ[IDCSCeh08768](https://cisco.com/cisco/webbugtool/bugdetails.do?bugID=IDCSCeh08768)が原因で使用率が高くなっている場合は、Cisco IOSソフトウェアリリース12.2(25)EWA2で解決できます。このプロセスはバックグラウンドプロセスであり、Catalyst 4500 スイッチの安定性には影響しません。



注：シスコの内部ツールおよび情報にアクセスできるのは、登録ユーザのみです。

<#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	0.02	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

トランク ポートの使用で IP Phone に接続されたときの高い CPU 使用率

ポートが音声 VLAN オプションとアクセス VLAN オプションの両方に対して設定されている場合、ポートはマルチ VLAN アクセスポートとして動作します。利点は、音声とアクセスの VLAN オプション用に設定された VLAN だけがトランクされることです。

電話にトランクされた VLAN は、STP インスタンスの数を増やします。スイッチは STP インスタンスを管理します。STP インスタンスにおける増加の管理によって、STP CPU 使用率も増加します。

すべての VLAN のトランキングも、不必要なブロードキャスト、マルチキャスト、未知のユニキャストトラフィックが電話リンクをヒットする原因になります。

<#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

RSPAN とレイヤ 3 コントロール パケットを伴う高い CPU 使用率

RSPAN でキャプチャされたレイヤ 3 コントロール パケットは、RSPAN 宛先インターフェイスだけではなく CPU 宛に送信され、それが高い CPU 使用率を引き起こします。L3 コントロール パケットは、CPU への転送を伴う静的な CAM エントリのアクションによってキャプチャされます。静的な CAM エントリは、すべての VLAN に対してグローバルです。不必要な CPU フラッディングを回避するには、Cisco IOS ソフトウェア リリース 12.2(37)SG 以降で利用可能な Per-VLAN Control Traffic Intercept 機能を使用します。

```
<#root>
```

```
Switch(config)#
```

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

静的 ACL は、入力 Feature TCAM の上にインストールされて、224.0.0.* の範囲内の既知の IP マルチキャスト アドレス宛に送信されるコントロール パケットをキャプチャします。静的 ACL は、ブート時にインストールされて、ユーザが設定した ACL よりも前に表示されます。静的 ACL は、常に最初にヒットされ、すべての VLAN 上で CPU へのコントロール トラフィックを傍受します。

Per-VLAN コントロール トラフィック傍受機能は、コントロール トラフィックをキャプチャする選択的な VLAN ごとのパス管理モードを提供します。入力 Feature TCAM の対応する静的 CAM エントリは、新しいモードで無効化されます。コントロール パケットは、スヌーピング機能やルーティング機能がイネーブルになった VLAN に接続された機能固有の ACL によってキャプチャされます。RSPAN VLAN に接続された機能固有の ACL はありません。そのため、RSPAN VLAN から受信されたすべてのレイヤ 3 コントロール パケットは CPU には転送されません。

CPU 宛のトラフィックを分析するためのツールのトラブルシューティング

このドキュメントで示したように、CPU を宛先とするトラフィックが、Catalyst 4500 の高い CPU 使用率の主な理由の 1 つです。CPU を宛先とするトラフィックは、設定による意図的なものか、設定ミスやサービス拒否攻撃による意図的ではないものいずれかです。CPU には、このトラフィックによるネットワークへの悪影響を防ぐために、QOS メカニズムが組み込まれています。ただし、CPU に送られるトラフィックの根本原因を特定し、トラフィックが希望とは異なるものである場合は、そのトラフィックを削除します。

ツール1:SPANでCPUトラフィックを監視する : Cisco IOSソフトウェアリリース12.1(19)EW以降

Catalyst 4500 では、標準の SPAN 機能を使用して、入力と出力の両方の CPU バウンド トラフィックを監視できます。宛先インターフェイスは、パケット スニファ ソフトウェアを実行するパケット モニタまたは管理者ラップトップに接続します。このツールにより、CPU が処理しているトラフィックを、迅速かつ正確に分析できます。このツールは、CPU パケットエンジンに送られる個別のキューを監視する機能を提供します。



注：スイッチングエンジンにはCPUトラフィック用に32のキューがあり、CPUパケットエンジンには16のキューがあります。

<#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
```

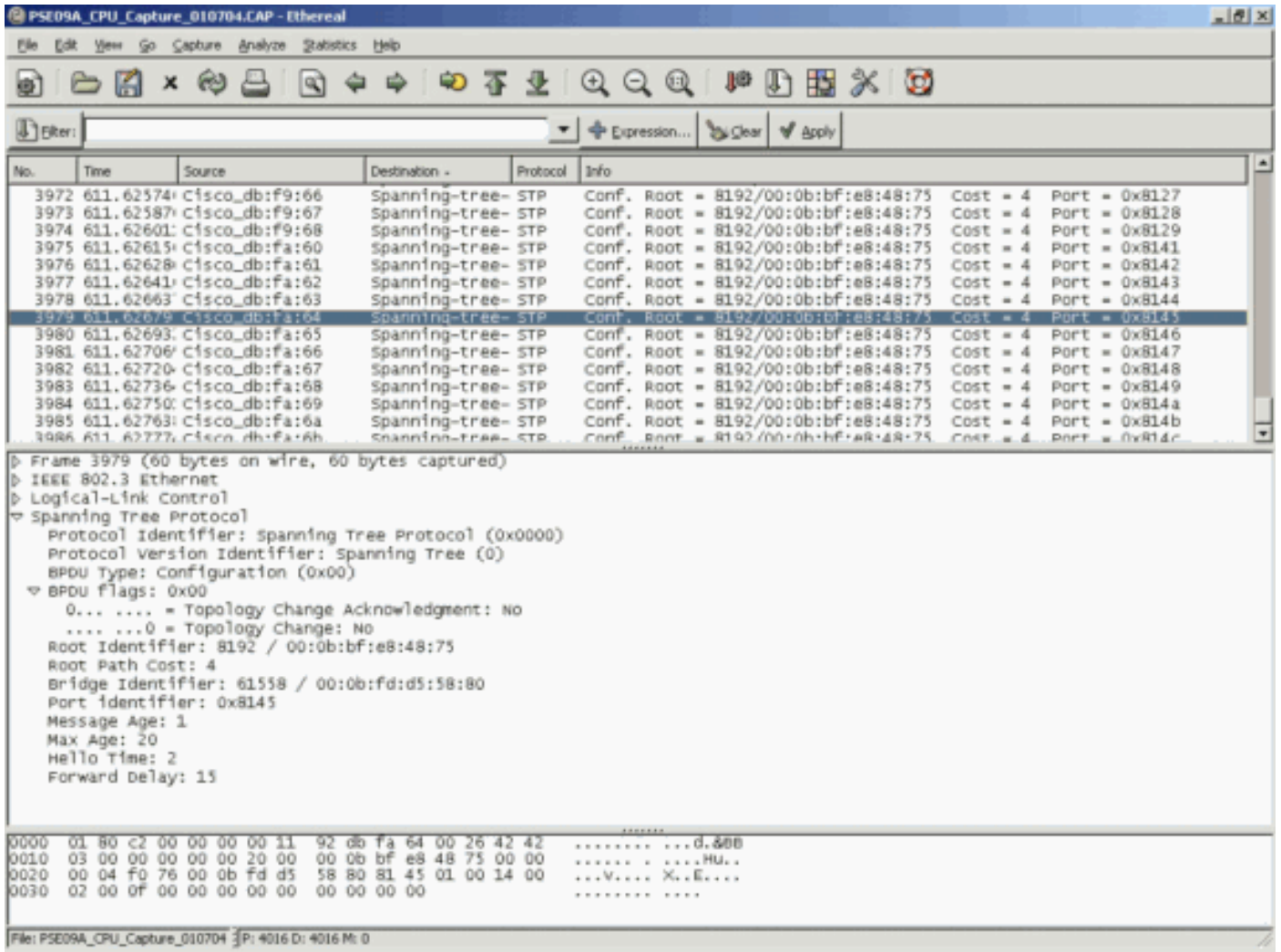
Session 1

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

スニファプログラムを実行する PC を接続した場合、トラフィックを即座に解析できます。このセクションのウィンドウで表示される出力には、高い CPU 使用率の原因が STP BPDU の過剰な数にあることを示しています。



注:CPUスニファのSTP BPDUは正常です。しかし、予想よりも多くの情報が表示される場合は、スーパーバイザエンジンの推奨制限を超えています。詳細は、このドキュメントの「大量のスパニングツリー ポート インスタンス」のセクションを参照してください。



CPU使用率が高いと、STP BPDUの数が過剰になる

ツール2：組み込みCPUスニファ：Cisco IOSソフトウェアリリース12.2(20)EW以降

Catalyst 4500 には、CPU を消費しているトラフィックを迅速に特定できる CPU スニファとデコーダが組み込まれています。このセクションの例に示すように、debug コマンドを使用してこのファシリティを有効にできます。この機能は、一度に 1024 パケットを維持できる循環バッファを実装しています。新しいパケットが到着すると、より古いパケットは上書きされます。CPU 使用率が高くなる問題をトラブルシューティングする際にこの機能を使用することは問題ありません。

```
<#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



注:debug コマンドを発行すると、CPU使用率が常に100 %近辺になります。debug コマンドを発行するときの高いCPU使用率になるのは普通のことです。

ツール3 : トラフィックをCPUに送信するインターフェイスの特定 : Cisco IOSソフトウェアリリース12.2(20)EW以降

Catalyst 4500 は、CPU 処理のためにトラフィック/パケットを送信するトップ インターフェイスを特定するためのもう 1 つの便利なツールを提供します。このツールによって、多数のブロードキャストや他のサービス拒否攻撃を CPU に送信するデバイスを即座に特定できます。CPU 使用率が高くなる問題をトラブルシューティングする際にこの機能を使用することは問題ありません。

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:debugコマンドを発行すると、CPU使用率が常に100 %近辺になります。debug コマンドを発行するとき高い CPU 使用率になるのは一般的なことです。

要約

Catalyst 4500 スイッチでは、高レート of IP バージョン 4 (IPv4) パケット転送をハードウェアで処理しています。一部の機能や例外によって、CPU が処理するパスを経由して一部のパケットが転送されることがあります。Catalyst 4500 は、CPU に送られるパケットの扱いには洗練された QoS メカニズムを使用しています。このメカニズムによって、スイッチの信頼性と安定性が確保され、また、同時に、パケットのソフトウェア転送のために CPU を最大化します。Cisco IOS ソフトウェア リリース 12.2(25)EWA2 以降は、アカウントティングと同様にパケット/プロセス処理の追加の機能強化も提供します。また、Catalyst 4500 には、CPU 使用率が高くなるシナリオの根本原因を特定するために役立つ十分なコマンドと、強力なツールが用意されています。しかし、ほとん

どの場合、Catalyst 4500 の高い CPU 使用率はネットワークを不安定にすることはなく、懸念する必要がないものです。

関連情報

- [「CPU Utilization on Catalyst 4500/4000, 2948G, 2980G, and 4912G Switches That Run CatOS Software \(Catalyst 4500/4000、2948G、2980G、および 4912G スイッチでの CPU 使用率の理解\)」](#)
- [LAN 製品に関するサポート ページ](#)
- [LAN スイッチングに関するサポート ページ](#)
- [テクニカル サポートとドキュメント - Cisco Systems](#)

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。