

Cisco IOS ソフトウェア ベースの Catalyst 4500 スイッチで CPU の使用率が高い

目次

[概要](#)

[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](#)

[CPU の使用率が高くなるその他の原因](#)

[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 : 組み込みの 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を特定する : Cisco IOS ソフトウェア](#)

[リリース 12.2\(20\)EW 以降](#)

[要約](#)

[関連情報](#)

概要

Catalyst 4948 スイッチなどの Catalyst 4500 シリーズ スイッチには、CPU に送られるトラフィック用の洗練されたパケット処理方式があります。一般に認識されている問題は、これらのスイッチの高い CPU 使用率です。この文書では、CPU のパケット処理アーキテクチャについて詳細に解説するとともに、これらのスイッチで CPU の使用率が高くなる原因を特定する方法について説明します。また、Catalyst 4500 シリーズの高い CPU 使用率の原因となる一般的なネットワークや設定のシナリオもいくつか示します。

注: Catalyst OS (CatOS) ベースの Catalyst 4500/4000 シリーズ スイッチを稼働している場合は、『[Catalyst 4000、2948G、2980G、および 4912G スイッチでの CPU 使用率の理解](#)』を参照してください。

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くものです。

- 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 ベースのスイッチだけに適用され、CatOS ベースのスイッチには適用されません。

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たものです。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。ネットワークが稼働中の場合は、コマンドが及ぼす潜在的な影響を十分に理解しておく必要があります。

表記法

ドキュメント表記の詳細は、『[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

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; ポート集約プロトコル) 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 カード電源装置ファントレイ
- 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。このメカニズムによって、レイヤ 2 (L2) /レイヤ 3 (L3) /レイヤ 4 (L4) パケット情報に基づく差別化を形成します。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

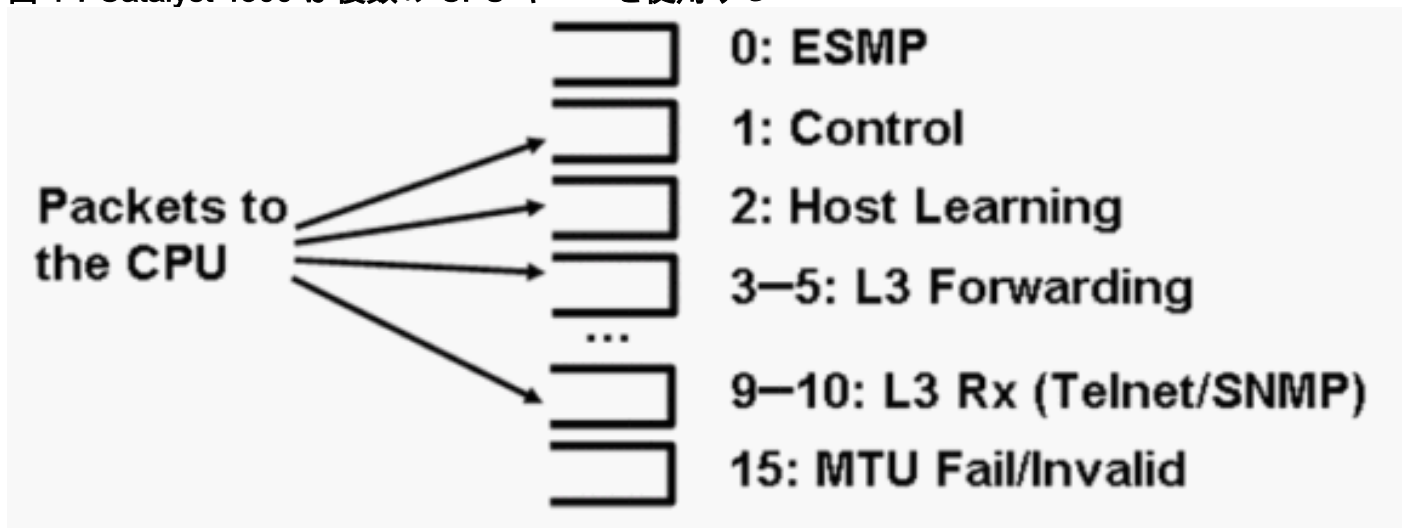


表 1 : Catalyst 4500 キューの説明

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 または他のコンポーネント管理のための ESMP1 パケット (内部管理パケット)

1	Control	STP、CDP、PAgP、LACP2、UDLD3 などの L2 コントロールプレーン パケット
2		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	ソフトウェアで転送する必要がある GRE ⁴ のようなパケットは ARP ⁵ が宛先 IP アドレスのために未解決なら、パケット送信されますトンネル伝送しますこのキューに。
6、7、8	L2 Fwd、 / L2 Fwd L2 Fwd	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。 IP ヘッダー オプション期限切れになった TTL7ARPA 以外のカプセル化
9、10	L3 Rx High、L3 Rx Low	L3 コントロールプレーントラフィックは、たとえば、ルーティング プロトコル、CPU IP アドレス例に向かう Telnet、SNMP および SSH ⁸ が含まれています。
11	RPF Failure	RPF9 チェックで障害があったマルチキャスト パケット
12	ACL fwd(snoping)	DHCP10 スヌーピング、ダイナミック ARP インスペクション、または IGMP11 スヌーピング機能によって処理されるパケット
13	ACL log, unreach	宛先へ出力 ACL の否定かルートの欠乏が廃棄された原因これらのパケットだった log キーワードまたはパケットとの ACE ¹² を押し込んだパケットは ICMP 到達不能 メッセージの生成を必要とします。
14	ACL sw processing	セキュリティ ACL に対する TCAM13 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	出力インターフェイス MTU サイズがパケッ

Fail/Invalid	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
--------------	----------------------------------

- 1 ESMP = Even Simple Management Protocol。
- 2 LACP = Link Aggregation Control Protocol。
- 3 UDLD = UniDirectional Link Detection (単方向リンク検出) 。
- 4 GRE = Generic Routing Encapsulation (総称ルーティング カプセル化) 。
- 5 ARP = Address Resolution Protocol (アドレス解決プロトコル) 。
- 6 SVI = switched virtual interface (スイッチ仮想インターフェイス) 。
- 7 TTL = Time to Live (存続可能時間) 。
- 8 SSH = Secure Shell Protocol (セキュア シェル プロトコル) 。
- 9 RPF = Reverse Path Forwarding (リバース パス転送)
- 10 DHCP = Dynamic Host Configuration Protocol。
- 11 IGMP = Internet Group Management Protocol (インターネット グループ管理プロトコル) 。
- 12 ACE = Access Control Entry (アクセス コントロール エントリ) 。
- 13 TCAM = Ternary Content Addressable Memory。

次のキューは独立したキューです。

- L2 Fwd Highest または L3 Fwd Highest
- L2 Fwd High/Medium または L3 Fwd High/Medium
- L2 Fwd Low または L3 Fwd Low
- L3 Rx High または L3 Rx Low

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、63 の DSCP のあるパケットは、L3 Fwd Highest キュー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 all** コマンド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。

```
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 Control 48 0 0 0 0 Host Learning 0 0 0 0 L3 Fwd High 0 0
0 0 L3 Fwd Medium 0 0 0 0 L3 Fwd Low 0 0 0 0 L2 Fwd High 0 0 0 0 L2 Fwd Medium 0 0 0 0
L2 Fwd Low 0 0 0 0 L3 Rx High 0 0 0 0 L3 Rx Low 0 0 0 0 RPF Failure 0 0 0 0 ACL
fwd(snooping) 0 0 0 0 ACL log, unreach 0 0 0 0 ACL sw processing 0 0 0 0 MTU Fail/Invalid
0 0 0 0 Packets Dropped by Packet Queue Queue Total 5 sec avg 1 min avg 5 min avg 1 hour avg -
----- Esmp 0 0 0 0
Control 0 0 0 0 Host Learning 0 0 0 0 L3 Fwd High 0 0 0 0 L3 Fwd Medium 0 0 0 0 L3 Fwd
Low 0 0 0 0 L2 Fwd High 0 0 0 0 L2 Fwd Medium 0 0 0 0 L2 Fwd Low 0 0 0 0 L3 Rx High 0 0
0 0 L3 Rx Low 0 0 0 0 RPF Failure 0 0 0 0 ACL fwd(snooping) 0 0 0 0 ACL log, unreach 0 0
0 0 ACL sw processing 0 0 0 0 MTU Fail/Invalid 0 0 0 0

```

Catalyst 4500 の CPU では、[表 1](#) の各キューに対して、重み付けを行っています。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ティングを強化しています。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](#)

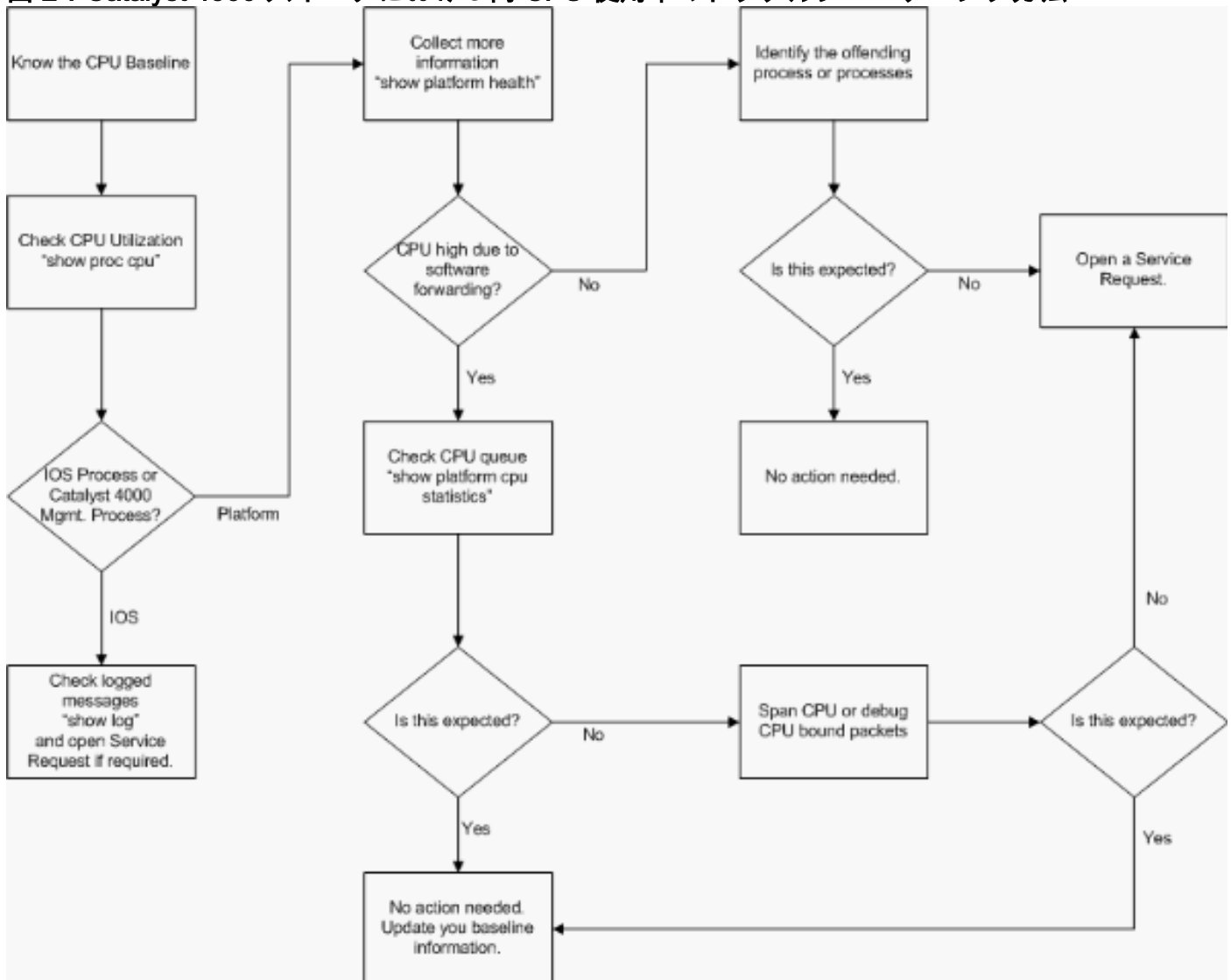
Catalyst 4500 の CPU パケット処理アーキテクチャと設計を理解してからでも、Catalyst 4500 の CPU 使用率が高い理由を特定したい方もいらっしゃるでしょう。Catalyst 4500 には、CPU の使用率が高くなる根本原因を特定するために必要なコマンドとツールが用意されています。理由を特定した後、管理者は次のいずれかの操作を実行できます。

- 修正操作：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解析を依頼するための [Cisco テクニカルサポート](#) サービス リクエストの作成などがあります。
- 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



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 health** 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** である場合、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 statistics** コマンドを発行します。このアクティビティが **K2CpuMan Review** プロセス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、このドキュメントと「[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ます。まだ根本原因を特定できない場合は、[Cisco テクニカルサポート](#)に問い合わせてください。

[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witchの CPU 使用率を調べるには、**show processes cpu** コマンドを実行します。CPU 使用率のベースラインを継続的に把握しておくことは、ネットワーク設定に別の設定を追加したときや、ネットワークトラフィックのパターンが変化したときなどに重要です。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。

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

5 秒間の CPU 使用率は次のように表記されます。

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

x% は CPU 使用率の合計を表し、y% は割り込みレベルで消費された CPU を表します。Catalyst 4500 スwitch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witchでの show processes cpu コマンドについて](#)

この **show processes cpu** の出力は、CPU を使用する 2 つのプロセス **Cat4k Mgmt HiPri** と **Cat4k 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 HiPri と **Cat4k 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 を使用するかを確認するには、**show platform health** コマンドを発行します。

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 cpu** コマンド出力は、**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 cpu** コマンド出力は、**Cat4k Mgmt LoPri** の下で追加の使用率をカウントし

ます。この Cat4k Mgmt LoPri 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。

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
S2w-JobEventSchedule	10.00	0.32	10	7	100	500	0	0	0	10:14
Stub-JobEventSchedul	10.00	12.09	10	6	100	500	14	13	9	396:35
StatValueMan Update	1.00	0.22	1	0	100	500	0	0	0	6:28
Pim-review	0.10	0.00	1	0	100	500	0	0	0	0:22
Ebm-host-review	1.00	0.00	8	0	100	500	0	0	0	0:05
Ebm-port-review	0.10	0.00	1	0	100	500	0	0	0	0:01
Protocol-aging-revie	0.20	0.00	2	0	100	500	0	0	0	0:00
Acl-Flattener e	1.00	0.00	10	0	100	500	0	0	0	0:00
KxAclPathMan create/	1.00	0.00	10	5	100	500	0	0	0	0:39
KxAclPathMan update	2.00	0.00	10	0	100	500	0	0	0	0:00
KxAclPathMan reprogr	1.00	0.00	2	0	100	500	0	0	0	0:00
TagMan-RecreateMtegR	1.00	0.00	10	0	100	500	0	0	0	0:00
K2CpuMan Review	30.00	10.19	30	28	100	500	14	13	9	397:11
K2AccelPacketMan: Tx	10.00	2.20	20	0	100	500	2	2	1	82:06
K2AccelPacketMan: Au	0.10	0.00	0	0	100	500	0	0	0	0:00
K2AclMan-taggedFlatA	1.00	0.00	10	0	100	500	0	0	0	0:00
K2AclCamMan stale en	1.00	0.00	10	0	100	500	0	0	0	0:00
K2AclCamMan hw stats	3.00	1.04	10	5	100	500	1	1	0	39:36
K2AclCamMan kx stats	1.00	0.00	10	5	100	500	0	0	0	13:40
K2AclCamMan Audit re	1.00	0.00	10	5	100	500	0	0	0	13:10
K2AclPolicerTableMan	1.00	0.00	10	1	100	500	0	0	0	0:38
K2L2 Address Table R	2.00	0.00	12	5	100	500	0	0	0	0:00
K2L2 New Static Addr	2.00	0.00	10	1	100	500	0	0	0	0:00
K2L2 New Multicast A	2.00	0.00	10	5	100	500	0	0	0	0:01
K2L2 Dynamic Address	2.00	0.00	10	0	100	500	0	0	0	0:00
K2L2 Vlan Table Revi	2.00	0.00	12	9	100	500	0	0	0	0:01
K2 L2 Destination Ca	2.00	0.00	10	0	100	500	0	0	0	0:00
K2PortMan Review	2.00	0.72	15	11	100	500	1	1	0	37:22
Gigaport65535 Review	0.40	0.07	4	2	100	500	0	0	0	3:38
Gigaport65535 Review	0.40	0.08	4	2	100	500	0	0	0	3:39
K2Fib cam usage revi	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib IrmFib Review	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib Vrf Default Ro	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib AdjRepop Revie	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib Vrf Unpunt Rev	2.00	0.01	15	0	100	500	0	0	0	0:23
K2Fib Consistency Ch	1.00	0.00	5	2	100	500	0	0	0	29:25
K2FibAdjMan Stats Re	2.00	0.30	10	4	100	500	0	0	0	6:21
K2FibAdjMan Host Mov	2.00	0.00	10	4	100	500	0	0	0	0:00
K2FibAdjMan Adj Chan	2.00	0.00	10	0	100	500	0	0	0	0:00
K2FibMulticast Signa	2.00	0.01	10	2	100	500	0	0	0	2:04
K2FibMulticast Entry	2.00	0.00	10	7	100	500	0	0	0	0:00
K2FibMulticast Irm M	2.00	0.00	10	7	100	500	0	0	0	0:00
K2FibFastDropMan Rev	2.00	0.00	7	0	100	500	0	0	0	0:00
K2FibPbr route map r	2.00	0.06	20	5	100	500	0	0	0	16:42
K2FibPbr flat acl pr	2.00	0.07	20	2	100	500	0	0	0	3:24
K2FibPbr consolidati	2.00	0.01	10	0	100	500	0	0	0	0:24
K2FibPerVlanPuntMan	2.00	0.00	15	4	100	500	0	0	0	0:00
K2FibFlowCache flow	2.00	0.01	10	0	100	500	0	0	0	0:23
K2FibFlowCache flow	2.00	0.00	10	0	100	500	0	0	0	0:00
K2FibFlowCache adj r	2.00	0.01	10	0	100	500	0	0	0	0:20
K2FibFlowCache flow	2.00	0.00	10	0	100	500	0	0	0	0:06

```

K2MetStatsMan Review      2.00  0.14    5    2 100 500    0  0  0 23:40
K2FibMulticast MET S      2.00  0.00   10    0 100 500    0  0  0 0:00
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
RkiosAclMan Review        3.00  0.06   30    0 100 500    0  0  0 2:35
MatMan Review             0.50  0.00    4    0 100 500    0  0  0 0:00
Slot 3 ILC Manager R      3.00  0.00   10    0 100 500    0  0  0 0:00
Slot 3 ILC S2wMan Re      3.00  0.00   10    0 100 500    0  0  0 0:00
Slot 4 ILC Manager R      3.00  0.00   10    0 100 500    0  0  0 0:00
Slot 4 ILC S2wMan Re      3.00  0.00   10    0 100 500    0  0  0 0:00
Slot 5 ILC Manager R      3.00  0.00   10    0 100 500    0  0  0 0:00
Slot 5 ILC S2wMan Re      3.00  0.00   10    0 100 500    0  0  0 0:00
Slot 6 ILC Manager R      3.00  0.00   10    0 100 500    0  0  0 0:00
Slot 6 ILC S2wMan Re      3.00  0.00   10    0 100 500    0  0  0 0:00
Slot 7 ILC Manager R      3.00  0.00   10    0 100 500    0  0  0 0:00
Slot 7 ILC S2wMan Re      3.00  0.00   10    0 100 500    0  0  0 0:00
EthHoleLinecardMan(1     1.66  0.04   10    0 100 500    0  0  0 1:18
EthHoleLinecardMan(2     1.66  0.02   10    0 100 500    0  0  0 1:18
EthHoleLinecardMan(6     1.66  0.17   10    6 100 500    0  0  0 6:38
-----
%CPU Totals                212.80  35.63

```

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

show platform health コマンドでは、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 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 actual 列で高い数値を探します。また、その列の右側を必ず見て、そのプロセスの CPU 使用状況の 1 分と 1 時間の average %CPU 列を確認します。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 health](#) コマンド出力では、Stub-JobEventSchedul プロセスと K2CpuMan Review プロセスが比較的多数の CPU サイクルを使用しています。表 2 に、show platform health コマンドで出力される、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をいくつか示します。

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
Acl-Flattener/K2AclMan	ACL マージ プロセス
KxAclPathMan - TagMan	ACL 状態管理とメンテナンス
K2CpuMan Review	プロセスはソフトウェア パケット転送を行うこのプロセスによる

	CPU使用率が高い状態を見る場合 show platform cpu packet statistics コマンドの使用の CPU を押す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
K2AclCamMan	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
K2AclPolicerTableMan	入出力ポリサーを管理します
K2L2	これらのプロセスによってさまざまな L2 表のメンテナンスに責任がある Catalyst 4500 Cisco IOS ソフトウェアの L2 フォワーディングサブシステムを表します。
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	FIB1 管理
K2FibFlowCache	PBR2 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	MET3 統計情報を管理します
K2QosDblMan 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 Aging Table 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- JobEventSchedule	ラインカード状態を監視するために S2W5 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	それぞれのラインカード内で GBIC6 を管理します

1 FIB = Forwarding Information Base (転送情報ベース) 。

2 PBR = Policy Based Routing (ポリシー ベース ルーティング) 。

3 MET = Multicast Expansion Table。

4 DBL = Dynamic Buffer Limiting。

5 S2W = serial-to-wire。

6 GBIC = 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い](#)
- [ICMP リダイレクト、同一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 のためのハードウェアリソース \(TCAM \) から](#)
- [ACL 内の log キーワード](#)
- [レイヤ 2 転送ループ](#)

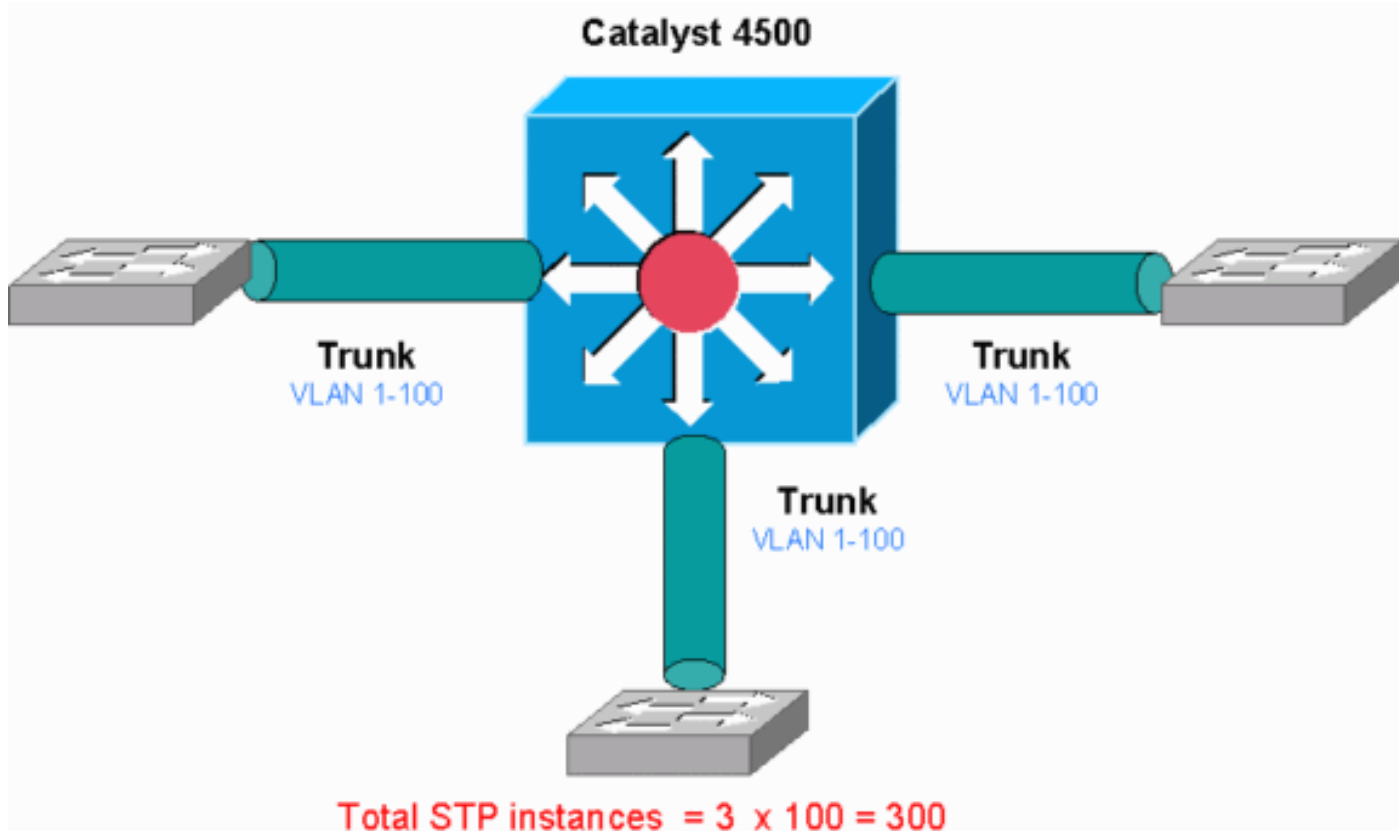
CPU へのパケット交換のその他の理由には次のものがあります。

- 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。
- **established** 以外の TCP フラグのある 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・スイッチングパス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 以降でのレート制限です。ACL のインターフェイスの下で **no ip unreachable** 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。



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。

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
S2w-JobEventSchedule	10.00	0.32	10	7	100	500	0	0	0	10:14
Stub-JobEventSchedul	10.00	12.09	10	6	100	500	14	13	9	396:35
StatValueMan Update	1.00	0.22	1	0	100	500	0	0	0	6:28
Pim-review	0.10	0.00	1	0	100	500	0	0	0	0:22
Ebm-host-review	1.00	0.00	8	0	100	500	0	0	0	0:05
Ebm-port-review	0.10	0.00	1	0	100	500	0	0	0	0:01
Protocol-aging-revie	0.20	0.00	2	0	100	500	0	0	0	0:00
Acl-Flattener e	1.00	0.00	10	0	100	500	0	0	0	0:00
KxAclPathMan create/	1.00	0.00	10	5	100	500	0	0	0	0:39
KxAclPathMan update	2.00	0.00	10	0	100	500	0	0	0	0:00
KxAclPathMan reprogr	1.00	0.00	2	0	100	500	0	0	0	0:00
TagMan-RecreateMtegR	1.00	0.00	10	0	100	500	0	0	0	0:00
K2CpuMan Review	30.00	10.19	30	28	100	500	14	13	9	397:11
K2AccelPacketMan: Tx	10.00	2.20	20	0	100	500	2	2	1	82:06
K2AccelPacketMan: Au	0.10	0.00	0	0	100	500	0	0	0	0:00
K2AclMan-taggedFlatA	1.00	0.00	10	0	100	500	0	0	0	0:00
K2AclCamMan stale en	1.00	0.00	10	0	100	500	0	0	0	0:00
K2AclCamMan hw stats	3.00	1.04	10	5	100	500	1	1	0	39:36
K2AclCamMan kx stats	1.00	0.00	10	5	100	500	0	0	0	13:40

K2AclCamMan Audit re	1.00	0.00	10	5	100	500	0	0	0	13:10
K2AclPolicerTableMan	1.00	0.00	10	1	100	500	0	0	0	0:38
K2L2 Address Table R	2.00	0.00	12	5	100	500	0	0	0	0:00
K2L2 New Static Addr	2.00	0.00	10	1	100	500	0	0	0	0:00
K2L2 New Multicast A	2.00	0.00	10	5	100	500	0	0	0	0:01
K2L2 Dynamic Address	2.00	0.00	10	0	100	500	0	0	0	0:00
K2L2 Vlan Table Revi	2.00	0.00	12	9	100	500	0	0	0	0:01
K2 L2 Destination Ca	2.00	0.00	10	0	100	500	0	0	0	0:00
K2PortMan Review	2.00	0.72	15	11	100	500	1	1	0	37:22
Gigaport65535 Review	0.40	0.07	4	2	100	500	0	0	0	3:38
Gigaport65535 Review	0.40	0.08	4	2	100	500	0	0	0	3:39
K2Fib cam usage revi	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib IrmFib Review	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib Vrf Default Ro	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib AdjRepop Revie	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib Vrf Unpunt Rev	2.00	0.01	15	0	100	500	0	0	0	0:23
K2Fib Consistency Ch	1.00	0.00	5	2	100	500	0	0	0	29:25
K2FibAdjMan Stats Re	2.00	0.30	10	4	100	500	0	0	0	6:21
K2FibAdjMan Host Mov	2.00	0.00	10	4	100	500	0	0	0	0:00
K2FibAdjMan Adj Chan	2.00	0.00	10	0	100	500	0	0	0	0:00
K2FibMulticast Signa	2.00	0.01	10	2	100	500	0	0	0	2:04
K2FibMulticast Entry	2.00	0.00	10	7	100	500	0	0	0	0:00
K2FibMulticast Irm M	2.00	0.00	10	7	100	500	0	0	0	0:00
K2FibFastDropMan Rev	2.00	0.00	7	0	100	500	0	0	0	0:00
K2FibPbr route map r	2.00	0.06	20	5	100	500	0	0	0	16:42
K2FibPbr flat acl pr	2.00	0.07	20	2	100	500	0	0	0	3:24
K2FibPbr consolidati	2.00	0.01	10	0	100	500	0	0	0	0:24
K2FibPerVlanPuntMan	2.00	0.00	15	4	100	500	0	0	0	0:00
K2FibFlowCache flow	2.00	0.01	10	0	100	500	0	0	0	0:23
K2FibFlowCache flow	2.00	0.00	10	0	100	500	0	0	0	0:00
K2FibFlowCache adj r	2.00	0.01	10	0	100	500	0	0	0	0:20
K2FibFlowCache flow	2.00	0.00	10	0	100	500	0	0	0	0:06
K2MetStatsMan Review	2.00	0.14	5	2	100	500	0	0	0	23:40
K2FibMulticast MET S	2.00	0.00	10	0	100	500	0	0	0	0:00
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
RkiosAclMan Review	3.00	0.06	30	0	100	500	0	0	0	2:35
MatMan Review	0.50	0.00	4	0	100	500	0	0	0	0:00
Slot 3 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 3 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 4 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 4 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 5 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 5 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 6 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 6 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 7 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 7 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
EthHoleLinecardMan(1	1.66	0.04	10	0	100	500	0	0	0	1:18
EthHoleLinecardMan(2	1.66	0.02	10	0	100	500	0	0	0	1:18
EthHoleLinecardMan(6	1.66	0.17	10	6	100	500	0	0	0	6:38

 %CPU Totals 212.80 35.63

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

。

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
S2w-JobEventSchedule	10.00	0.32	10	7	100	500	0	0	0	10:14
Stub-JobEventSchedul	10.00	12.09	10	6	100	500	14	13	9	396:35
StatValueMan Update	1.00	0.22	1	0	100	500	0	0	0	6:28
Pim-review	0.10	0.00	1	0	100	500	0	0	0	0:22
Ebm-host-review	1.00	0.00	8	0	100	500	0	0	0	0:05
Ebm-port-review	0.10	0.00	1	0	100	500	0	0	0	0:01
Protocol-aging-revie	0.20	0.00	2	0	100	500	0	0	0	0:00
Acl-Flattener e	1.00	0.00	10	0	100	500	0	0	0	0:00
KxAclPathMan create/	1.00	0.00	10	5	100	500	0	0	0	0:39
KxAclPathMan update	2.00	0.00	10	0	100	500	0	0	0	0:00
KxAclPathMan reprogr	1.00	0.00	2	0	100	500	0	0	0	0:00
TagMan-RecreateMtegR	1.00	0.00	10	0	100	500	0	0	0	0:00
K2CpuMan Review	30.00	10.19	30	28	100	500	14	13	9	397:11
K2AccelPacketMan: Tx	10.00	2.20	20	0	100	500	2	2	1	82:06
K2AccelPacketMan: Au	0.10	0.00	0	0	100	500	0	0	0	0:00
K2AclMan-taggedFlatA	1.00	0.00	10	0	100	500	0	0	0	0:00
K2AclCamMan stale en	1.00	0.00	10	0	100	500	0	0	0	0:00
K2AclCamMan hw stats	3.00	1.04	10	5	100	500	1	1	0	39:36
K2AclCamMan kx stats	1.00	0.00	10	5	100	500	0	0	0	13:40
K2AclCamMan Audit re	1.00	0.00	10	5	100	500	0	0	0	13:10
K2AclPolicerTableMan	1.00	0.00	10	1	100	500	0	0	0	0:38
K2L2 Address Table R	2.00	0.00	12	5	100	500	0	0	0	0:00
K2L2 New Static Addr	2.00	0.00	10	1	100	500	0	0	0	0:00
K2L2 New Multicast A	2.00	0.00	10	5	100	500	0	0	0	0:01
K2L2 Dynamic Address	2.00	0.00	10	0	100	500	0	0	0	0:00
K2L2 Vlan Table Revi	2.00	0.00	12	9	100	500	0	0	0	0:01
K2 L2 Destination Ca	2.00	0.00	10	0	100	500	0	0	0	0:00
K2PortMan Review	2.00	0.72	15	11	100	500	1	1	0	37:22
Gigaport65535 Review	0.40	0.07	4	2	100	500	0	0	0	3:38
Gigaport65535 Review	0.40	0.08	4	2	100	500	0	0	0	3:39
K2Fib cam usage revi	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib IrmFib Review	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib Vrf Default Ro	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib AdjRepop Revie	2.00	0.00	15	0	100	500	0	0	0	0:00
K2Fib Vrf Unpunt Rev	2.00	0.01	15	0	100	500	0	0	0	0:23
K2Fib Consistency Ch	1.00	0.00	5	2	100	500	0	0	0	29:25
K2FibAdjMan Stats Re	2.00	0.30	10	4	100	500	0	0	0	6:21
K2FibAdjMan Host Mov	2.00	0.00	10	4	100	500	0	0	0	0:00
K2FibAdjMan Adj Chan	2.00	0.00	10	0	100	500	0	0	0	0:00
K2FibMulticast Signa	2.00	0.01	10	2	100	500	0	0	0	2:04
K2FibMulticast Entry	2.00	0.00	10	7	100	500	0	0	0	0:00
K2FibMulticast Irm M	2.00	0.00	10	7	100	500	0	0	0	0:00
K2FibFastDropMan Rev	2.00	0.00	7	0	100	500	0	0	0	0:00
K2FibPbr route map r	2.00	0.06	20	5	100	500	0	0	0	16:42
K2FibPbr flat acl pr	2.00	0.07	20	2	100	500	0	0	0	3:24
K2FibPbr consolidati	2.00	0.01	10	0	100	500	0	0	0	0:24
K2FibPerVlanPuntMan	2.00	0.00	15	4	100	500	0	0	0	0:00
K2FibFlowCache flow	2.00	0.01	10	0	100	500	0	0	0	0:23
K2FibFlowCache flow	2.00	0.00	10	0	100	500	0	0	0	0:00
K2FibFlowCache adj r	2.00	0.01	10	0	100	500	0	0	0	0:20
K2FibFlowCache flow	2.00	0.00	10	0	100	500	0	0	0	0:06
K2MetStatsMan Review	2.00	0.14	5	2	100	500	0	0	0	23:40
K2FibMulticast MET S	2.00	0.00	10	0	100	500	0	0	0	0:00
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
RkiosAclMan Review	3.00	0.06	30	0	100	500	0	0	0	2:35
MatMan Review	0.50	0.00	4	0	100	500	0	0	0	0:00
Slot 3 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 3 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 4 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 4 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 5 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 5 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 6 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 6 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 7 ILC Manager R	3.00	0.00	10	0	100	500	0	0	0	0:00
Slot 7 ILC S2wMan Re	3.00	0.00	10	0	100	500	0	0	0	0:00
EthHoleLinecardMan(1	1.66	0.04	10	0	100	500	0	0	0	1:18
EthHoleLinecardMan(2	1.66	0.02	10	0	100	500	0	0	0	1:18
EthHoleLinecardMan(6	1.66	0.17	10	6	100	500	0	0	0	6:38

 %CPU Totals 212.80 35.63

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 LoPri と Spanning 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。

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>!---</i> 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
<i>!---</i> 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、K2CpuMan Review プロセスという CPU に送られるパケットを処理するジョブが CPU を使い切っていることがわかります。

Switch#show platform health

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

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。

```
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
```

[ステップ 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をチェックできます。出力が根本原因を明確に特定します。

```
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もかなり低く下げするために、必要のない VLAN をトランクから手動でプルーニングします。

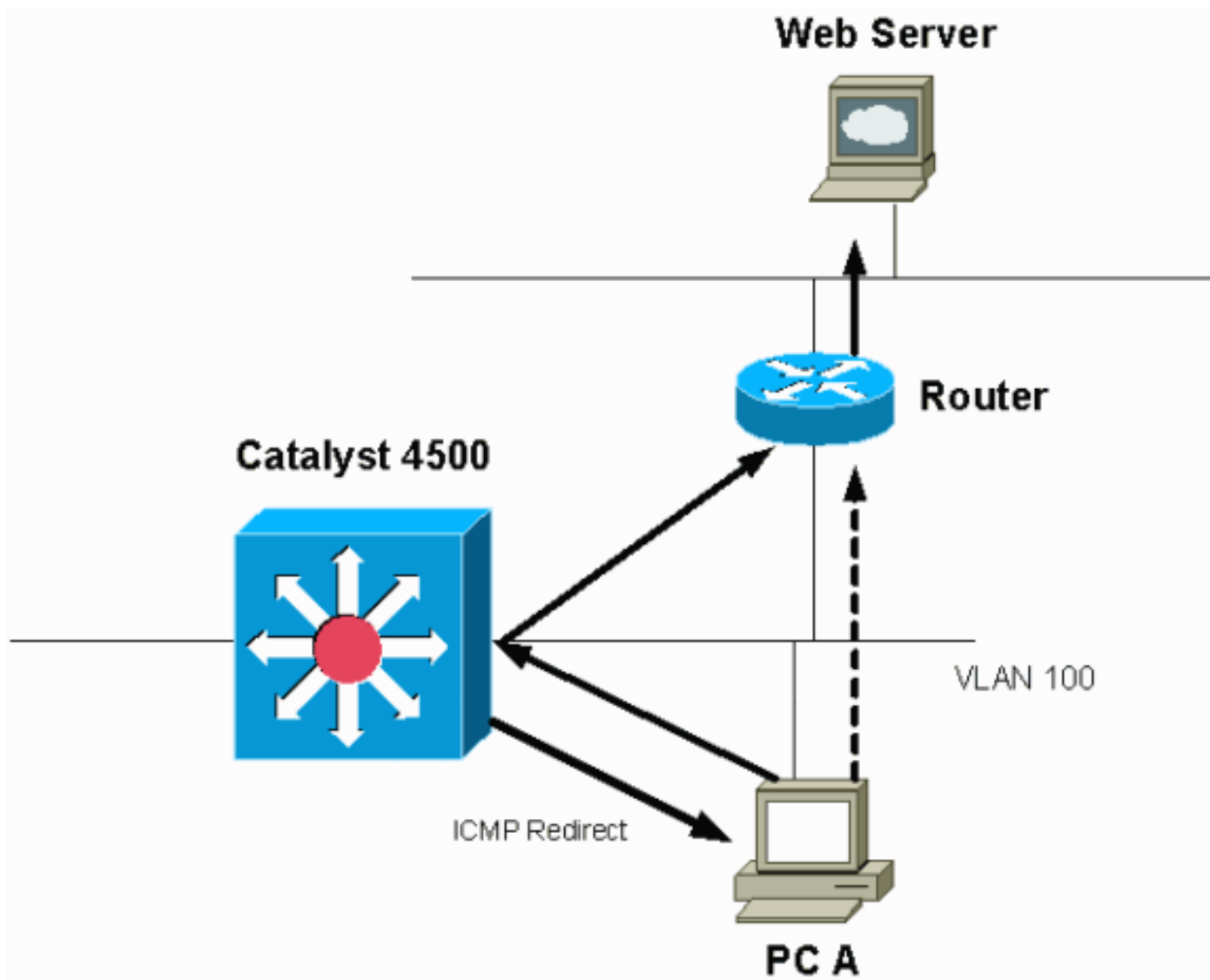
ヒント：IP Phone のポートをトランク ポートとして設定しないように注意してください。これはよく起こる設定ミスです。IP Phone ポートは音声 VLAN 設定で設定します。この設定は、擬似トランクを作成しますが、不必要な VLAN を手動でプルーニングする必要はありません。音声ポートの設定方法の詳細については、『[音声インターフェイスの設定](#)』ソフトウェア設定ガイドを参照してください。Cisco 以外の IP Phone は、この音声 VLAN や補助 VLAN 設定をサポートしません。シスコ以外の IP phone を接続しているポートは、手作業で削除する必要があります。

[ICMP リダイレクト、同一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 インターフェイスの入出力のトラフィックは、スイッチによる ICMP リダイレクトに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。これをディセーブルにするには、`no 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 LoPri と IP 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。

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

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。

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

手順 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 Low キューが非常に多くのトラフィックを受信していることがわかります。

```
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 -----
Esmpl 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
L3 Rx High 257147 0 0 0
L3 Rx Low 5325772 10 19 13 7
RPF Failure 155 0 0 0
ACL fwd(snooping) 65604591 53 54 54 53
ACL log, unreach 11013420 9 8 8 8
```

ステップ 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](#)」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。

送信元のデバイスが、Catalyst 4500 が送信する ICMP リダイレクトに対して動作し、宛先へのネクスト ホップを変更することを期待する場合があります。しかし、すべてのデバイスが ICMP リダイレクトに応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。インターフェイスの下で `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。

この「[ICMP リダイレクト、同一インターフェイス上のルーティング パケット](#)」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経由する 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: 同一の VLAN 内での IPX および AppleTalk トラフィックのスイッチングは、プロセスのスイッチングを必要としません。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。

```
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。

```
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
```


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。

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。

```
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 -----
Esmpl 48613268 38 39 38 39 Control 142166648 74 74 73 73 Host Learning 1845568 2 2 2 2 L3 Fwd
High 17 0 0 0 0 L3 Fwd Medium 2626 0 0 0 0 L3 Fwd Low 1582414 1 1 1 1 L2 Fwd Medium 1 0 0 0 0 L2
Fwd Low          576905398      1837      1697      1938      1515
L3 Rx High          257147          0          0          0          0
L3 Rx Low           5325772          10         19         13         7
RPF Failure           155            0            0            0            0
ACL fwd(snooping)  65604591         53          54          54          53
ACL log, unreach   11013420          9            8            8            8
```

ステップ 4: 根本原因を特定します。

管理者が IPX と AppleTalk のルーティングを設定しているため、根本原因は簡単に特定できるはずですが、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](#)」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 エンジンが、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。

```
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
```

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。

```
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 : 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。

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。

```
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 -----
Esmpr 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
L3 Fwd Medium 2626 0 0 0
L3 Fwd Low 1582414 1 1 1 1
L2 Fwd Medium 1 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
```

ステップ 4 : 根本原因を特定します。

show platform health コマンドの出力によって、CPU が数多くの新しい MAC アドレスを確認することがわかります。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の変更は、十分に検討した場合にだけ行ってください。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、ACL エントリの数が過剰に設定されると、TCAM の容量を超えることとなります。表 3 は、Catalyst 4500 Supervisor Engine とスイッチのそれぞれで利用できる TCAM リソースの数を示しています。

表 3 : Catalyst 4500 スーパーバイザ エンジンおよびスイッチの TCAM 容量

製品	Feature TCAM (方向ごと)	QoS TCAM (方向ごと)
Supervisor Engine II+/II+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 のマスクが付いている 16,384 のエントリ	2048 のマスクが付いている 16,384 のエントリ
Supervisor Engine V-10GE および Catalyst 4948-10GE	16,384 のマスクが付いている 16,384 のエントリ	16,384 のマスクが付いている 16,384 のエントリ

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。

```
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 -----
Esmpr 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

このエラーメッセージは、**show logging** コマンドの出力に現れます。最終的に、このメッセージは、ソフトウェア処理が開始され、その結果として CPU 使用率が高くなる可能性があることを示します。

注: 大きな ACL を変更した場合、このメッセージを簡単に確認してから、変更された ACL を TCAM でもう一度プログラムします。

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。

show processes cpu コマンドを発行します。Cat4k Mgmt LoPri プロセスが CPU サイクルのほとんどを占有するので、CPU 使用率が高いことがわかります。

```
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。

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

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` コマンドを実行します。 ACL sw processing キューが大量の packets を受信していることが分かります。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。

```
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: NVRAM をフォーマットします。

[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ことがあります。トラフィック検出段階を除き、log キーワード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には log キーワードは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、log メッセージにはレート制限があります。ACL に一致するパケットの数をカウントするために log 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 log メッセージのレビューが ACL ロギング機能の仕様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` 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。

```
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 コマンドで 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 Review** プロセスがほとんどの CPU サイクルを使用していることに注目してください。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。

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

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。

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。このコマンド出力では、パケットの受信が **ACL log** キーワードによるものであることを確認できます。

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

Packets Dropped by Packet Queue

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

ステップ 4 : NVRAM をフォーマットします。

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。この問題を避けるために、ACL から **log** キーワードを削除します。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。 **show access-list acl_id** コマンド出力でアクセス リスト カウンタを確認できます。

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 LoPri と Spanning 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。

```
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、K2CpuMan Review プロセスという CPU に送られるパケットを処理するジョブが CPU を使い切っていることがわかります。

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

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。

```
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
```

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：根本原因を特定して問題を解決します。

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）。

1. ループを識別します。
2. ループの範囲を検出します。
3. ループを遮断します。
4. ループの原因を修正します。
5. 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 ソフトウェアを稼働する 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 機能を実装します。

- **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 ガード機能拡張](#)』を参照してください。
- **UDLD**：単方向リンクを検出する、転送ループを防ぎます。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 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](#)
- 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。

```
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、KxAclPathMan create プロセスが 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。

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

ステップ 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-status コマンドを発行します。次の例が示すように、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 range コマンドを使用できます。

```
Switch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#interface range gigabitethernet 5/1 - 48
Switch(config-if-range)#logging event link-status
Switch(config--if-range)#end
Switch#show logging
```

```
!--- Output suppressed. 3w5d: %LINK-3-UPDOWN: Interface GigabitEthernet5/24, changed state to
down 3w5d: %LINK-3-UPDOWN: Interface GigabitEthernet5/24, changed state to up 3w5d: %LINK-3-
UPDOWN: Interface GigabitEthernet5/24, changed state to down 3w5d: %LINK-3-UPDOWN: Interface
GigabitEthernet5/24, changed state to up 3w5d: %LINK-3-UPDOWN: Interface GigabitEthernet5/24,
changed state to down 3w5d: %LINK-3-UPDOWN: Interface GigabitEthernet5/24, changed state to up
```

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、インターフェ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 health コマンドの出力は、**K2Fib Consistency Ch** がほとんどの CPU 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 分間または 1 時間以上経過してもわずかであり、チェックが定期的な短い確認であることが分かります。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。

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

K2FibAdjMan Host Move プロセスでの高い CPU 使用率

Catalyst 4500 では、**K2FibAdjMan Host Move** プロセスの CPU 使用率が高く表示されることがあります。この高い使用率は **show platform health** 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り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 タイムを増やすが、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 ID [CSCed15021](#) (登録ユーザ専用) を参照してください。

```
Switch#show platform health
          %CPU   %CPU   RunTimeMax   Priority   Average %CPU   Total
          Target Actual Target Actual     Fg   Bg 5Sec Min Hour   CPU
```

```

Lj-poll                1.00  0.02    2    1  100  500    0  0  0  1:09
GalChassisVp-review   3.00  0.29   10    3  100  500    0  0  0  11:15
S2w-JobEventSchedule 10.00  0.32   10    7  100  500    0  0  0  10:14
!--- Output suppressed. K2FibAdjMan Stats Re 2.00 0.30 10 4 100 500 0 0 0 6:21 K2FibAdjMan Host
Mov  2.00 18.68   10    4  100  500   25 29  28 2134:39
K2FibAdjMan Adj Chan  2.00  0.00   10    0  100  500    0  0  0  0:00
K2FibMulticast Signa  2.00  0.01   10    2  100  500    0  0  0  2:04
K2FibMulticast Entry  2.00  0.00   10    7  100  500    0  0  0  0:00

```

グローバル設定モードでは、MAC アドレスの最大エージング タイムを変更できます。コマンド構文はルータの場合は `mac-address-table aging-time seconds` であり、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](#)』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 `show platform health` コマンドの出力で `RkiosPortMan Port Review` プロセスの高い CPU 使用率を表示することがあります。Cisco bug ID [CSCeh08768](#) (登録ユーザ専用) は、高い使用率の原因であり、これは Cisco IOS ソフトウェア リリース 12.2(25)EWA2 で解決されています。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。

```

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

```

[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。

```

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フラッシ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。

```
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

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

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せん。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。

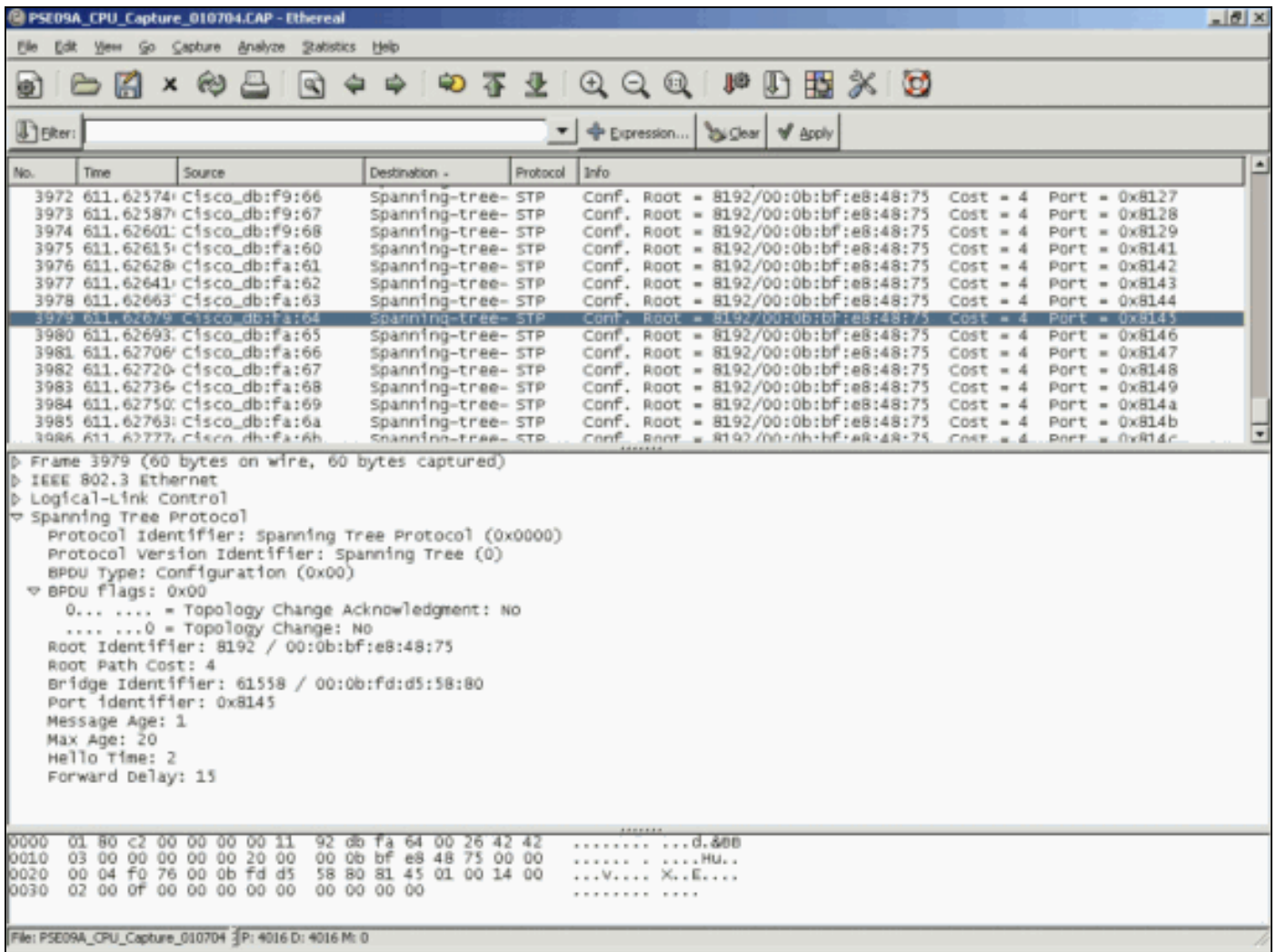
注: スイッチング エンジンには、CPU トラフィック用に 32 個のキューがあり、CPU パケット エンジンには 16 個のキューがあります。

```
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 : Gil/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 スニファ内の STP BPDU は正常です。しかし、予想よりも多くのものが見つかった場合、スーパーバイザ エンジン用の推奨制限を超えている可能性があります。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。



ツール 2 : 組み込みの 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が示すように、**debug** コマンドでこのファシリティをイネーブルにできます。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。

```
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高い CPU 使用率になるのは一般的なことです。

[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を特定する：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のではなく、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](#)