

# Catalyst 9000スイッチでの出カドロップについて

## 内容

---

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[確認](#)

[影響を受けるインターフェイスの特定](#)

[着信インターフェイスと発信インターフェイスの特定](#)

[バッファ割り当て](#)

[トラブルシューティング](#)

[バッファ割り当ての変更](#)

[キュー単位バッファの変更](#)

[Wiresharkによる出カドロップの分析](#)

[代替アプローチ](#)

[関連情報](#)

---

## はじめに

このドキュメントでは、UADP ASICベースのCatalyst 9000シリーズプラットフォームで高速インターフェイスの出カドロップをトラブルシューティングする方法について説明します。

## 前提条件

### 要件

次の項目に関する知識があることが推奨されます。

- 標準的なQoSの概念
- モジュラQoSコマンドラインインターフェイス(CLI)
- Wireshark

### 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。

- UADP 2.0およびUADP 3.0 ASICタイプ
- Catalyst 9200

- Catalyst 9300
- Catalyst 9400
- Catalyst 9500
- Catalyst 9600
- Cisco IOS® XE 16.Xまたは17.Xソフトウェア

 注：シスコの他のプラットフォームでこれらの機能を有効にするために使用されるコマンドについては、該当するコンフィギュレーション ガイドを参照してください。

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。

## 背景説明

高速インターフェイスでの出力廃棄は、あらゆるネットワーク環境で発生する問題であり、特に 10 Gbps以上のデータ転送速度をサポートするインターフェイスを扱う場合に問題となります。出力のドロップは、パケットがネットワークに送信される前にインターフェイスによってドロップされると発生します。

出力廃棄が使用率の低いレベルで発生した場合、インターフェイス使用率がどのように解釈されるかは誤解されがちです。

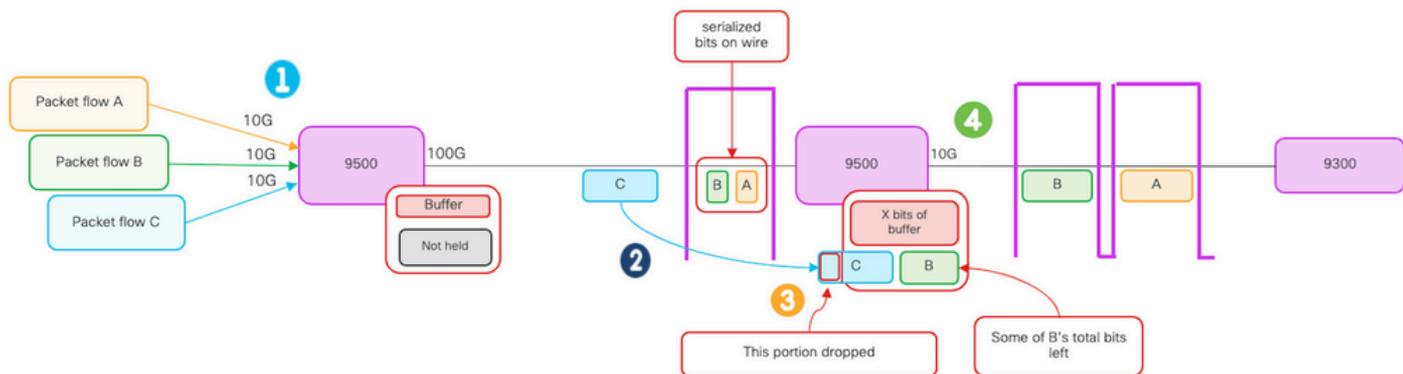
- 1つのインターフェイスから送信されるトラフィックが多すぎると、そのインターフェイスのバッファが過負荷になり、出力廃棄が発生する可能性があります。インターフェイスのバッファが少なすぎると、出力ドロップの可能性も高くなります。
- 出力ドロップは、インターフェイスの速度と接続デバイスの速度の不一致によっても発生する可能性があります。たとえば、10 Gbpsのインターフェイスが1 Gbpsのデータ転送速度だけをサポートするデバイスに接続されている場合、デバイスの機能に合わせてデータレートを低下させることができない場合、そのインターフェイスはパケットをドロップできます。

しかし、ほとんどの場合、出力廃棄は、ポートバッファを使い果たしたマイクロバーストトラフィックによって引き起こされます。

- マイクロバーストとは、大量のトラフィックが非常に短時間で送信される現象を指し、通常は数ミリ秒しか続かず、インターフェイスの使用可能な帯域幅を超えます。
- マイクロバーストが発生すると、ネットワークデバイスはトラフィックがネットワークに送信できるようになるまでトラフィックをバッファリングする必要があります。バッファサイズを超えると、パケットは廃棄されます。

ネットワークトラフィックは、通常、リンクの平均使用率で測定されます（設定に応じて30秒から5分で測定）。この平均は安定した比較的均一なフローを示していますが、ミリ秒のスケールで見ると、インターフェイス使用率は非常に高いバースト性を持つことが多くなります。

図 1. は、高速インターフェイスでの出力廃棄の根本的な原因を視覚的に示しています。



1. パケットフローA、B、およびCは、フロー内のビットのグループを表します。
2. フローA、B、およびCは100Gインターフェイスを介して送信され、ネクストホップスイッチで受信されます。
3. Aおよび一部のBビットが10G出力インターフェイスから正常に転送されることを確認します。
  - ただし、フローCが到着すると、到着と同時にビットをデキューできないため、スイッチはフローBの一部のビットをバッファリングしました。
  - バッファがいっぱいになり、パケットCの残りのBビットとすべてのビットを保持できなくなります。
  - 入力分類子が次のフレームの1ビットにもスペースがないと判断すると、パケット全体が廃棄されます。
4. フローAとBの一部が10Gインターフェイスを介して送信されます。

インターフェイスの「速度/帯域幅」は両方ともある程度誤った呼び方をしています。

- 帯域幅=データ (ビット) / 時間
- ビットが10G > 100G > 10Gの範囲を通過するとき、ビットは「物理的に速度が上がる」ことはない

「速度」の違いは、インターリーピング機能、レーン数、時間間隔当たりのパルス数、符号化メカニズムなどです。メディア (光/電子) の速度は高くなります。

 ヒント：ロードインターバルの遅延を秒単位で変更するには、インターフェイスコンフィギュレーションモードでload-interval <30-600>コマンドを使用します。(ロード間隔は、スイッチがインターフェイスカウンタをポーリングする頻度を表します)。

## 確認

高速インターフェイスでの出力ドロップのトラブルシューティングは複雑なプロセスになる可能性があります。問題の特定と解決に役立つ一般的な手順を次に示します。

影響を受けるインターフェイスを特定します。

- 最初に、出力廃棄が発生しているインターフェイスを絞り込みます。
- CLIでshow interfaces | include is up|Total output dropsコマンドの出力に表示されるか、監視ツールを使用して、問題が発生しているインターフェイスを判別します。

着信インターフェイスと発信インターフェイスの特定:

- インターフェイスからASICへのマッピングを確認するには、`show platform software fed <switch|active> ifm mappings`コマンドを実行します。

バッファ割り当ての確認:

- 影響を受けるインターフェイスのバッファ割り当てとインターフェイス設定を確認することが重要です。

Wiresharkでマイクロバーストを確認します。

- 高速インターフェイスでの出力廃棄は、多くの場合、トラフィックのマイクロバーストによって発生します。
- Wiresharkなどのトラフィック分析ツールを使用して、トラフィックパターンを監視し、マイクロバーストの可能性を特定します。

ハードウェアのアップグレードを検討します。

- 前の手順で問題が解決しない場合は、インターフェイス、デバイス、ネットワークインフラストラクチャなどのハードウェアをアップグレードして、増加したトラフィックを処理する必要があります。

## 影響を受けるインターフェイスの特定

出カドロップが発生している影響を受けるインターフェイスを特定するには、`show interfaces`コマンドを使用します。

- このコマンドは、入力エラーと出力エラー、廃棄されたパケット、およびその他の重要な情報に関する統計情報を含む、各インターフェイスに関する詳細情報を提供します。

インターフェイスのリストを絞り込み、影響を受けるインターフェイスをすばやく特定するには、`show interfaces | include is up|Total output drops`コマンドの出力。インターフェイスのダウンまたはadmin downをフィルタリングし、アクティブでドロップのあるインターフェイスのみを表示します。

- たとえば、このコマンドを使用すると、出力廃棄が発生したインターフェイスだけを表示できます。

```
<#root>
```

```
Cat9k(config)#
```

```
show interfaces | in is up|Total output drops
```

```
HundredGigE1/0/1 is up, line protocol is up (connected)  
Input queue: 0/2000/0/0 (size/max/drops/flushes);
```

```
Total output drops: 54845
```

```
HundredGigE1/0/10 is up, line protocol is up (connected)
Input queue: 0/2000/0/0 (size/max/drops/flushes);

Total output drops: 1540231
```

--snip--

---

 ヒント:show interfacesコマンドを使用し、該当するインターフェイスをすばやく簡単に特定するために適切な基準で出力をフィルタリングします。必要な手順を実行して問題を解決します。

---

Catalyst 9000シリーズスイッチでは、デフォルトで、出力パケットドロップはパケットではなくバイトで表示されます。検出された出力廃棄の量が実際に影響を与えたのか、あるいは単に一時的なバーストトラフィックが原因なのかを判断することが重要です。

インターフェイスで送信された出力バイトのうち、廃棄されたものの合計のパーセンテージを計算するには、次のコマンドを実行します。

- インターフェイスから出カドロップ数の合計を収集します。
- インターフェイスから出力バイト数の合計を収集します。
- 出力廃棄の割合を計算します。出カドロップ数の合計を出力バイト数の合計で除算し、100を乗算します。

これは、インターフェイスで廃棄された出力バイトのパーセンテージを示します。これは、対処が必要な輻輳やバッファ割り当ての問題があるかどうか、または出カドロップが一時的なマイクロバーストトラフィックによって発生したかどうかを判断するのに役立ちます。

showinterface <interface>コマンドを使用して、情報を収集します。

```
<#root>
```

```
Cat9k#
```

```
show interfaces twentyFiveGigE 1/0/41
```

```
TwentyFiveGigE1/0/41 is up, line protocol is up (connected)
Hardware is Twenty Five Gigabit Ethernet, address is dc77.4c8a.4289 (bia dc77.4c8a.4289)
MTU 1500 bytes, BW 25000000 Kbit/sec, DLY 10 usec,
    reliability 255/255, txload 3/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
```

```
Full-duplex, 10Gb/s
```

```
, link type is auto, media type is SFP-10GBase-AOC1M
input flow-control is on, output flow-control is off
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:06, output 00:00:10, output hang never
```

```
Last clearing of "show interface" counters 6w1d
```

Input queue: 0/2000/0/0 (size/max/drops/flushes);

Total output drops: 299040207

Queueing strategy: Class-based queueing

Output queue: 0/40 (size/max)

30 second input rate 767000 bits/sec, 155 packets/sec

30 second output rate 14603000 bits/sec, 1819 packets/sec

931864194 packets input, 572335285416 bytes, 0 no buffer

Received 933005 broadcasts (933005 multicasts)

0 runts, 0 giants, 0 throttles

0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored

0 watchdog, 0 multicast, 0 pause input

0 input packets with dribble condition detected

1067891106 packets output,

5930422327799

bytes,

0 underruns

--snip--

出力廃棄の合計 : 299040207

合計出力バイト数 : 5930422327799

出力廃棄のパーセンテージ =  $299040207 / 5930422327799 \times 100 = 0.005\%$

この例では、過去6週間にこのインターフェイスで送信された総バイト数の0.005 %が出力ドロップ数です ( カウンタ6w1dの最後のクリア )。

- パケットの合計数とドロップされた数の差は小さく、影響はありません。

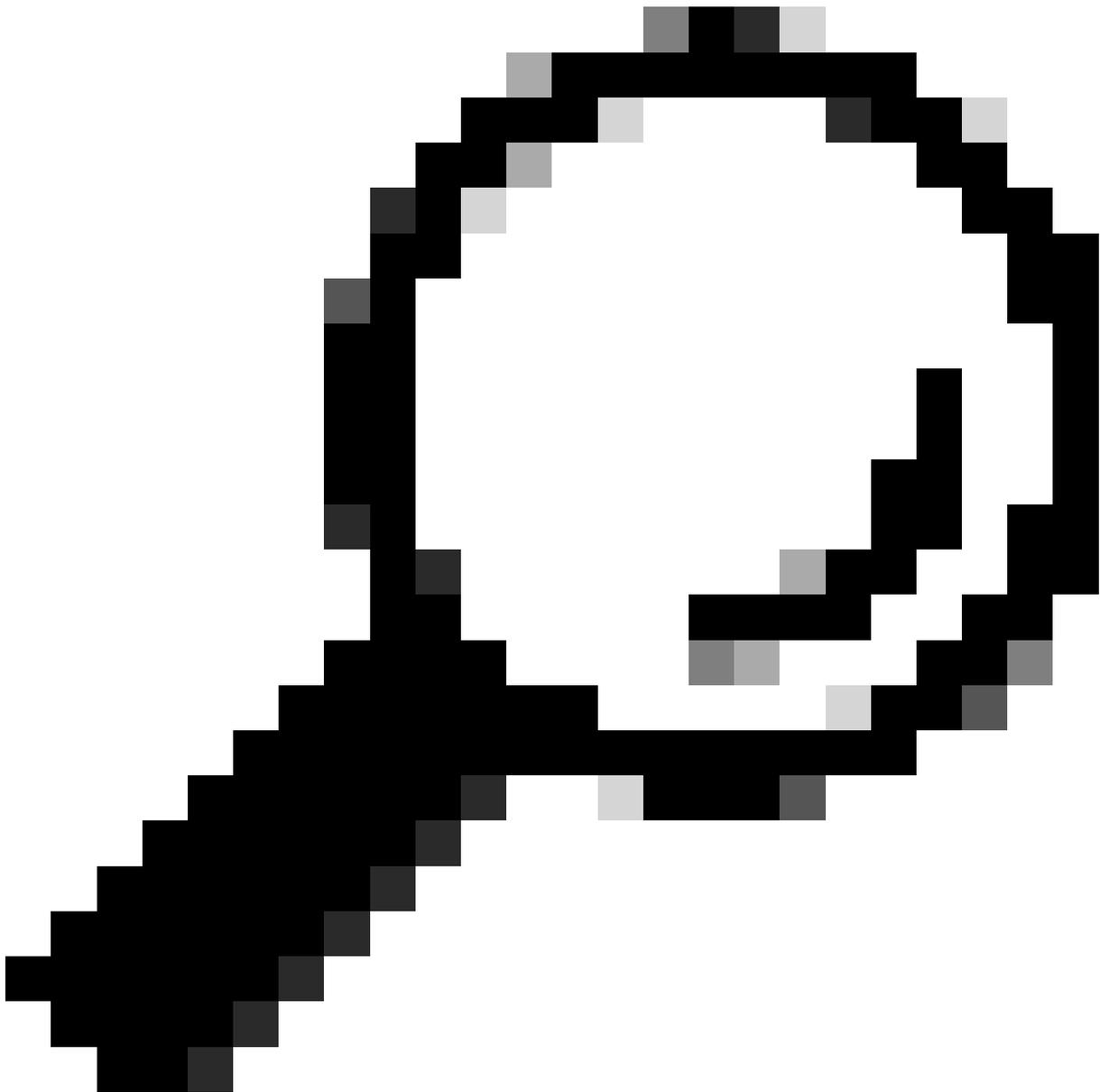
## 着信インターフェイスと発信インターフェイスの特定

Catalyst 9000シリーズスイッチで柔軟なバッファをより適切に割り当て、トラフィックを管理するには、異なるASICの着信インターフェイスと発信インターフェイスの選択を検討してください。

。

動的バッファまたは共有バッファとも呼ばれる柔軟なバッファは、輻輳またはトラフィック負荷の高い期間にパケットを一時的に保存するために動的に割り当てられるメモリの部分を指します。

。



ヒント:Catalyst 9000シリーズスイッチでのバッファ割り当ての詳細については、『[Catalyst 9000スイッチのキューバッファ割り当てについて](#)』を参照してください。

---

Catalyst 9000スイッチの特定モデルのアーキテクチャに基づいて、さまざまな機能を担う複数のASICが組み込まれていることがよくあります。

インターフェイスからASICへのマッピングを確認するには、`show platform software fed <switch|active> ifm mappings`コマンドを実行します。

この例では、インターフェイスからASICへのマッピングを示しています。インターフェイス範囲 `TenGigabitEthernet1/0/1 ~ TenGigabitEthernet1/0/24`はASIC 1にマッピングされ、残りはASIC 0にマッピングされます。

<#root>



1	11	0	0	11	12	12	NIF	Y
TenGigabitEthernet1/0/13 0x15 2								
1								
0	12	0	11	0	13	13	NIF	Y
TenGigabitEthernet1/0/14 0x16 2								
1								
0	13	0	10	1	14	14	NIF	Y
TenGigabitEthernet1/0/15 0x17 2								
1								
0	14	0	9	2	15	15	NIF	Y
TenGigabitEthernet1/0/16 0x18 2								
1								
0	15	0	8	3	16	16	NIF	Y
TenGigabitEthernet1/0/17 0x19 2								
1								
0	16	0	7	4	17	17	NIF	Y
TenGigabitEthernet1/0/18 0x1a 2								
1								
0	17	0	6	5	18	18	NIF	Y
TenGigabitEthernet1/0/19 0x1b 2								
1								
0	18	0	5	6	19	19	NIF	Y
TenGigabitEthernet1/0/20 0x1c 2								
1								
0	19	0	4	7	20	20	NIF	Y
TenGigabitEthernet1/0/21 0x1d 2								
1								
0	20	0	3	8	21	21	NIF	Y
TenGigabitEthernet1/0/22 0x1e 2								
1								
0	21	0	2	9	22	22	NIF	Y
TenGigabitEthernet1/0/23 0x1f 2								
1								
0	22	0	1	10	23	23	NIF	Y
TenGigabitEthernet1/0/24 0x20 2								
1								
0	23	0	0	11	24	24	NIF	Y
TenGigabitEthernet1/0/25 0x21 1								
0								
1	24	0	11	0	25	25	NIF	Y
TenGigabitEthernet1/0/26 0x22 1								
0								

1	25	0	10	1	26	26	NIF	Y
TenGigabitEthernet1/0/27 0x23 1								
0								
1	26	0	9	2	27	27	NIF	Y
TenGigabitEthernet1/0/28 0x24 1								
0								
1	27	0	8	3	28	28	NIF	Y
TenGigabitEthernet1/0/29 0x25 1								
0								
1	28	0	7	4	29	29	NIF	Y
TenGigabitEthernet1/0/30 0x26 1								
0								
1	29	0	6	5	30	30	NIF	Y
TenGigabitEthernet1/0/31 0x27 1								
0								
1	30	0	5	6	31	31	NIF	Y
TenGigabitEthernet1/0/32 0x28 1								
0								
1	31	0	4	7	32	32	NIF	Y
TenGigabitEthernet1/0/33 0x29 1								
0								
1	32	0	3	8	33	33	NIF	Y
TenGigabitEthernet1/0/34 0x2a 1								
0								
1	33	0	2	9	34	34	NIF	Y
TenGigabitEthernet1/0/35 0x2b 1								
0								
1	34	0	1	10	35	35	NIF	Y
TenGigabitEthernet1/0/36 0x2c 1								
0								
1	35	0	0	11	36	36	NIF	Y
TenGigabitEthernet1/0/37 0x2d 0								
0								
0	36	0	11	11	37	37	NIF	Y
TenGigabitEthernet1/0/38 0x2e 0								
0								
0	37	0	10	10	38	38	NIF	Y
TenGigabitEthernet1/0/39 0x2f 0								
0								
0	38	0	9	9	39	39	NIF	Y
TenGigabitEthernet1/0/40 0x30 0								
0								

```

0 39 0 8 8 40 40 NIF Y
TenGigabitEthernet1/1/1 0x31 0
0
0 40 0 0 19 41 41 NIF N
TenGigabitEthernet1/1/2 0x32 0
0
0 41 0 0 18 42 42 NIF N
TenGigabitEthernet1/1/3 0x33 0
0
0 42 0 0 17 43 43 NIF N
TenGigabitEthernet1/1/4 0x34 0
0
0 43 0 0 16 44 44 NIF N
TenGigabitEthernet1/1/5 0x35 0
0
0 44 0 0 15 45 45 NIF N
TenGigabitEthernet1/1/6 0x36 0
0
0 45 0 0 14 46 46 NIF N
TenGigabitEthernet1/1/7 0x37 0
0
0 46 0 0 13 47 47 NIF N
TenGigabitEthernet1/1/8 0x38 0
0
0 47 0 0 12 48 48 NIF N
FortyGigabitEthernet1/1/1 0x39 0
0
0 48 0 4 4 49 49 NIF N
FortyGigabitEthernet1/1/2 0x3a 0
0
0 49 0 0 0 50 50 NIF N

```

## バッファ割り当て

バッファは、輻輳やその他の変数が原因で転送できないトラフィックを一時的に保存するために使用されるため、バッファの割り当ては出力ドロップを回避するための重要な要素です。使用可能なバッファが十分でない場合、トラフィックは廃棄され、ネットワークのパフォーマンスが低下し、中断が発生する可能性があります。この検証により、高いトラフィック負荷を処理するのに十分なバッファスペースがスイッチにあることを確認できます。

show platform hardware fed switch active qos queue stats interface <interface>コマンドを使用すると、インターフェイスのキューごとの統計情報が表示されます。この統計情報には、バッファ

にキューイングされたバイト数、および使用可能なバッファの不足によって廃棄されたバイト数などが含まれます。

この例では、

- キュー0 ~ 4は現在、キューに入れられたトラフィックを保持しています。このインターフェイスで受信されたトラフィックは、送信可能になるまで一時的にバッファに保存されます。
- 廃棄または廃棄されたトラフィック(24010607バイト)が発生しているのはキュー2だけです。

<#root>

Cat9k#s

```
how platform hardware fed active qos queue stats interface twentyFiveGigE 1/0/41
```

-----  
AQM Global counters

GlobalHardLimit: 16257 | GlobalHardBufCount: 0

GlobalSoftLimit: 39039 | GlobalSoftBufCount: 0  
-----

High Watermark Soft Buffers: Port Monitor Disabled  
-----

Asic:0 Core:0 DATA Port:8 Hardware Enqueue Counters  
-----

Q Buffers

Enqueue-TH0

Enqueue-TH1

Enqueue-TH2

	(Count)	Qpolicer	(Bytes)	(Bytes)	(Bytes)	(Bytes)
0	0		0	40588200	9368282	0
1	0		0	23584521	789524	0
2	0		0	0	110307150901	0
3	0		0	0	487852543	0
4	0		0	0	5483512	0
5	0		0	0	0	0
6	0		0	0	0	0
7	0		0	0	0	0

Asic:0 Core:0 DATA Port:8 Hardware Drop Counters  
-----

Q

Drop-TH0

Drop-TH1

Drop-TH2

	SBufDrop (Bytes)	QebDrop (Bytes)	QpolicerDrop (Bytes)	(Bytes)	(By
0	0	0	0	0	
1	0	0	0	0	
2					
	0	0			
24010607					
<-- (drops on Q2)	0	0	0		
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	0	

## トラブルシューティング

### バッファ割り当ての変更

インターフェイスで使用されるソフトバッファの値を増やすには、グローバルコンフィギュレーションモードで `qos queue-softmax-multiplier` コマンドを使用します。

- 100 ~ 4800の範囲で値を指定します。デフォルト値は 100 です。
- このコマンドにより、マイクロバーストを吸収する単一のポートキューの機能が向上します。
- このコマンドを実行すると、ポートキューのしきい値が増加し、ポートキューが共有プールから追加のバッファユニットを消費できるようになります。

次の設定は、すべてのインターフェイスに適用されます。

- バッファ割り当て自体は、マイクロバーストがスイッチ上のすべてのポートで同時に発生するわけではありません。
- マイクロバーストがランダムな瞬間に発生した場合、共有バッファは、それらを吸収するために追加のバッファユニットを専用に割り当てることができます。

ソフトバッファの割り当てを変更するには、グローバルコンフィギュレーションモードで `qos queue-softmax-multiplier<100 4800>` コマンドを使用します。使用可能な最大値に設定すると、出力廃棄を排除または削減できる可能性が最も高くなります。これは、可能な限り廃棄を避けるために、一般的に推奨されるベストプラクティスです。

<#root>

Cat9k(config)#

qos queue-softmax-multiplier ?

<100-4800> multiplier(%)

show platform hardware fed active qos queue config interface <interface>コマンドを使用して、Catalyst 9000シリーズのソフトバッファの割り当てを確認します。

次の例は、Catalyst 9500で10 Gbpsの速度にネゴシエートされたインターフェイスに割り当てられているデフォルトのソフトバッファを示しています。

<#root>

Cat9k#

show platform hardware fed active qos queue config interface twentyFiveGigE 1/0/41

```

Asic:0 Core:0 DATA Port:8 GPN:141 LinkSpeed:0x12
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 64 - 71
  DrainFast:Disabled PortSoftStart:5 - 4320 BufferSharing:Disabled
    DTS Hardmax Softmax PortSMin GblSMin PortStEnd QEnable
  -----
  0  1  6  480  8

```

1920

```

  16  960  0  0  3  5760  En

```

<-- 1920 is the total soft buffers allocated to queue 0 on interface twentyFiveGigE 1/0/41

```

  1  1  5  0  11

```

2880

```

  16  1440  8  720  3  5760  En

```

<-- 2880 is the total soft buffers allocated to queue 1 on interface twentyFiveGigE 1/0/41

```

  2  1  5  0  6  0  0  0  0  0  3  5760  En
  3  1  5  0  6  0  0  0  0  0  3  5760  En
  4  1  5  0  6  0  0  0  0  0  3  5760  En
  5  1  5  0  6  0  0  0  0  0  3  5760  En
  6  1  5  0  6  0  0  0  0  0  3  5760  En
  7  1  5  0  6  0  0  0  0  0  3  5760  En

```

Priority	Shaped/shared	weight	shaping_step	sharpedWeight
0	0	Shared	50	0
1	0	Shared	75	0
2	0	Shared	10000	0
3	0	Shared	10000	0
4	0	Shared	10000	0
5	0	Shared	10000	0
6	0	Shared	10000	0
7	0	Shared	10000	0

Port	Port	Port	Port
Priority	Shaped/shared	weight	shaping_step
2	Shaped	1023	1023
QPolicer	Refresh Credit	Max Credit	Interval Idx

```

0 Disabled          0          0          0
1 Disabled          0          0          0
2 Disabled          0          0          0
3 Disabled          0          0          0
4 Disabled          0          0          0
5 Disabled          0          0          0
6 Disabled          0          0          0
7 Disabled          0          0          0

```

```

Weight0 Max_Th0 Min_Th0 Weigth1 Max_Th1 Min_Th1 Weight2 Max_Th2 Min_Th2
-----

```

```
0 0
```

```
1912
```

```
0 0
```

```
2137
```

```
0 0
```

```
2400
```

```
0
```

```
<-- Thresholds values in queue 0 on interface twentyFiveGigE 1/0/41
```

```
1 0
```

```
2295
```

```
0 0
```

```
2565
```

```
0 0
```

```
2880
```

```
0
```

```
<-- Thresholds values in queue 1 on interface twentyFiveGigE 1/0/41
```

```

2 0 0 0 0 0 0 0 0 0
3 0 0 0 0 0 0 0 0 0
4 0 0 0 0 0 0 0 0 0
5 0 0 0 0 0 0 0 0 0
6 0 0 0 0 0 0 0 0 0
7 0 0 0 0 0 0 0 0 0

```

次の例は、4800マルチプライヤが設定されたCatalyst 9500で10 Gbpsの速度にネゴシエートされたインターフェイスに割り当てられたソフトバッファを示しています。

```
<#root>
```

```
Cat9k#
```

```
show platform hardware fed active qos queue config interface twentyFiveGigE 1/0/41
```

```

Asic:0 Core:0 DATA Port:8 GPN:141 LinkSpeed:0x12
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 64 - 71
  DrainFast:Disabled PortSoftStart:4 - 42000 BufferSharing:Disabled
  DTS Hardmax Softmax PortSMin G1b1SMin PortStEnd QEnable

```

-----  
0 1 6 480 10

42000

1 1312 0 0 4 42000 En

<-- 42000 is the total soft buffers allocated to queue 0 on interface twentyFiveGigE 1/0/41

1 1 5 0 10

42000

1 1312 1 1312 4 42000 En

<-- 42000 is the total soft buffers allocated to queue 1 on interface twentyFiveGigE 1/0/41

2	1	5	0	6	0	0	0	0	0	4	42000	En
3	1	5	0	6	0	0	0	0	0	4	42000	En
4	1	5	0	6	0	0	0	0	0	4	42000	En
5	1	5	0	6	0	0	0	0	0	4	42000	En
6	1	5	0	6	0	0	0	0	0	4	42000	En
7	1	5	0	6	0	0	0	0	0	4	42000	En

Priority	Shaped/shared	weight	shaping_step	shapedWeight
----------	---------------	--------	--------------	--------------

0	0	Shared	50	0	0
1	0	Shared	75	0	0
2	0	Shared	10000	0	0
3	0	Shared	10000	0	0
4	0	Shared	10000	0	0
5	0	Shared	10000	0	0
6	0	Shared	10000	0	0
7	0	Shared	10000	0	0

Port	Port	Port	Port
Priority	Shaped/shared	weight	shaping_step

QPolicer	Shaped Refresh Credit	1023 Max Credit	1023 Interval	Idx
0 Disabled	0	0	0	0
1 Disabled	0	0	0	0
2 Disabled	0	0	0	0
3 Disabled	0	0	0	0
4 Disabled	0	0	0	0
5 Disabled	0	0	0	0
6 Disabled	0	0	0	0
7 Disabled	0	0	0	0

Weight0 Max\_Th0 Min\_Th0 Weigth1 Max\_Th1 Min\_Th1 Weight2 Max\_Th2 Min\_Th2

0 0

33851

0 0

37833

0 0

42480

0

<-- Thresholds values in queue 0 on interface twentyFiveGigE 1/0/41

```

1      0
33468
      0      0
37406
      0      0
42000
      0
<-- Thresholds values in queue 1 on interface twentyFiveGigE 1/0/41
2      0      0      0      0      0      0      0      0      0
3      0      0      0      0      0      0      0      0      0
4      0      0      0      0      0      0      0      0      0
5      0      0      0      0      0      0      0      0      0
6      0      0      0      0      0      0      0      0      0
7      0      0      0      0      0      0      0      0      0

```

 注：柔軟なバッファの割り当ては変化します。割り当てが上記の出力と一致しない可能性があります。割り当て出力は、使用されている特定のプラットフォーム、適用されているQoSポリシー、および対象のインターフェイスの動作速度のネゴシエートに応じて異なります。

## キュー単位バッファの変更

SoftMax乗数を使用できないシナリオ、またはトラフィックプロファイルに合わせてバッファを微調整しようとするシナリオでは、キュー単位のバッファ変更を利用できます。

- キューバッファの割り当てをインターフェイスごとに変更するには、ポリシーマップを使用する必要があります。
- ほとんどの場合、インターフェイスの現在のポリシーマップを変更し、バッファをクラス単位で変更します。

この例のインターフェイスでは、twentyFiveGigE 1/0/1で出力廃棄が発生しています。コマンド出力に示されているように、このインターフェイスに適用される出力ポリシーマップ。

show platform hardware fed switch active qos queue stats interface <interface>コマンドを使用すると、インターフェイスのキューごとの統計情報が表示されます。この統計情報には、バッファにキューイングされたバイト数や、使用可能なバッファの不足によって廃棄されたバイト数などが含まれます。

```
<#root>
```

```
Cat9k#s
```

```
how platform hardware fed active qos queue stats interface twentyFiveGigE 1/0/1
```

-----

```

AQM Global counters
GlobalHardLimit: 16257 | GlobalHardBufCount: 0
GlobalSoftLimit: 39039 | GlobalSoftBufCount: 0

```

```

-----
High Watermark Soft Buffers: Port Monitor Disabled
-----

```

```

Asic:0 Core:0 DATA Port:8 Hardware Enqueue Counters
-----

```

Q	Buffers (Count)	Enqueue-TH0 (Bytes)	Enqueue-TH1 (Bytes)	Enqueue-TH2 (Bytes)	Qpolicer (Bytes)
0	0	0	0	82	0
1	0	0	0	7517	0
2	0	0	0	110307150901	0
3	0	0	0	7174010710	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0

```

Asic:0 Core:0 DATA Port:8 Hardware Drop Counters
-----

```

Q	Drop-TH0 (Bytes)	Drop-TH1 (Bytes)	Drop-TH2 (Bytes)	SBufDrop (Bytes)	Qebl (By
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
24010607					
3	0	0	0	0	
20071103					
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	0	

Enqueueカウンタに基づいて、このインターフェイスでの出力廃棄を軽減するために、Q0からQ1の間のエンキューレートは非常に低く、したがってQ2およびQ3ほど多くのバッファを必要としません。キュー2とキュー3には、他のキューよりも大量のトラフィックがキューイングされるので、より多くのバッファをキュー2とキュー3に割り当てることを推奨します。

- バッファ割り当てを変更するには、インターフェイスtwentyFiveGigE 1/0/1に適用されるポリシーマップでqueue-buffers ratio <0-100>設定を使用します。



注：このコマンドをポリシーの各クラスに設定する場合、合計が100になる必要があります。ただし、1つのクラスだけが設定されている場合、システムは他のキューから均等にバッファを差し引きます。

---

次の例は、ポリシーマップでqueue-buffers ratioを設定する方法を示しています。

```
<#root>
Cat9k(config)#
policy-map test

Cat9k(config-pmap)#
class Voice

Cat9k(config-pmap-c)#
priority level 1
```

```

Cat9k(config-pmap-c)#
queue-buffers ratio 5

Cat9k(config-pmap-c)#
class Video

Cat9k(config-pmap-c)#
bandwidth remaining percent 50

Cat9k(config-pmap-c)#
queue-buffers ratio 15

Cat9k(config-pmap-c)#
class BuisnessCritical

Cat9k(config-pmap-c)#
bandwidth remaining percent 30

Cat9k(config-pmap-c)#
queue-buffers ratio 40          <-- Queue 3

Cat9k(config-pmap-c)#
class class-default

Cat9k(config-pmap-c)#
bandwidth remaining percent 20

Cat9k(config-pmap-c)#
queue-buffers ratio 40          <-- Queue 4

```

Cisco IOS XE 17.2.1リリースからは、UADP 3.0(Catalyst 9500 High PerformanceおよびCatalyst 9600)に基づくスイッチは、同じASIC内の2つのコア間でActive Queue Management(AQM)バッファを共有するように設定できます。

- バッファ共有が設定されたポートでは、AQMバッファがマップされているコアに関係なく、使用可能な任意のAQMバッファが使用されます。
- これは、単一のAQMコアのバッファを飽和させたトラフィックの高バーストの管理に役立ちます。
- この機能をイネーブルにするには、グローバルコンフィギュレーションモードでqos share-bufferコマンドを使用します。
- show platform hardware fed active qos queue config interfaceコマンドで、機能を確認でき

ます。これは、システム全体に影響を与えるグローバル設定です。

バッファ共有を無効にするには、no qos share-bufferコマンドのno形式を使用します。

```
<#root>
Cat9k(config)#
qos share-buffer

Cat9k(config)#
end
```

## Wiresharkによる出力ドロップの分析

ネットワーク上のマイクロバーストの存在を確認するには、Wiresharkなどのパケットキャプチャツールを使用できます。

- Wiresharkはパケットをキャプチャし、リアルタイムまたはキャプチャ後にネットワークトラフィックを分析できるようにします。
- Wiresharkでドロップが発生したときにどのようなマイクロバーストが発生するかを特定するには、影響を受けるインターフェイスでパケットキャプチャを開始し、出力のドロップが発生するまで繰り返しインターフェイスを確認します。
- キャプチャが完了したら、Wiresharkでキャプチャファイルを開き、トラフィック量が多い期間が続いてトラフィック量が少ない期間または少ない期間がないかを確認します。これらはマイクロバーストの可能性ががあります。

インターフェイスでの出力ドロップを効果的にキャプチャして分析するには、次の推奨事項を使用します。

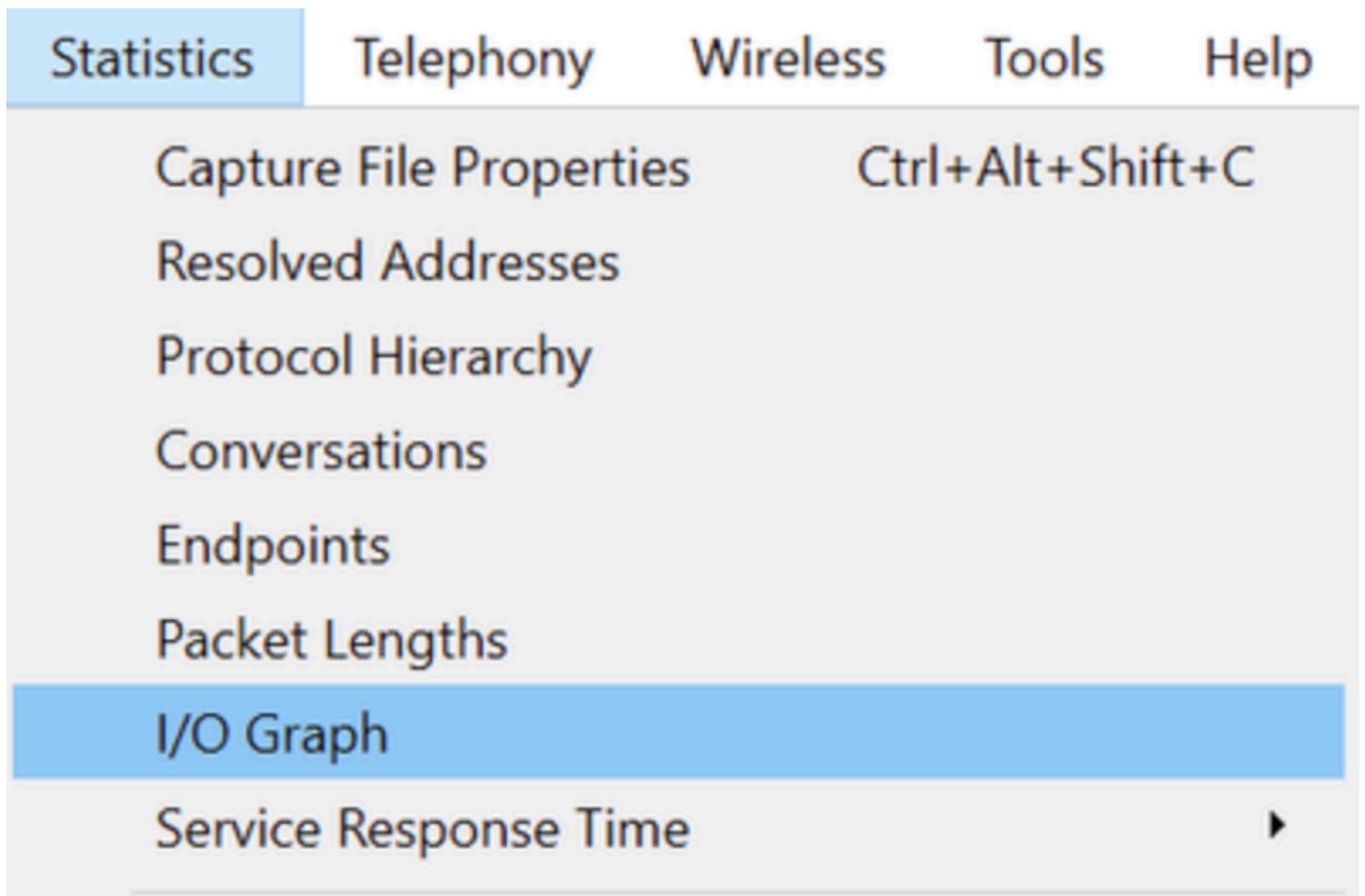
- EPC(Embedded Packet Capture)は使用しないでください。EPCはすべてのトラフィックを最大1000パケット/秒(pps)に制限するため、データが無効になる可能性があります。
- 出力ドロップが特定されるインターフェイスのTX専用SPANが推奨される方法です。
- SPANを使用する場合は、送信元ポートと宛先ポートの速度が同じであるか、または宛先ポートの速度が送信元ポートよりも高速である必要があります。これにより、SPANセッションで、ポート速度の不一致による追加のネットワーク輻輳やパケットドロップが発生しなくなります。
- 出力廃棄がアクティブに増加している間にSPANセッションを収集することが重要です。これにより、関連するトラフィックをキャプチャし、マイクロバーストの根本原因を特定できます。ドロップが発生した後にSPANセッションが収集されると、関連するトラフィックをキャプチャできなくなります。

このような大量のトラフィックがマイクロバーストとして発生しているかどうかを確認するには、WiresharkのI/Oグラフ機能を使用します。I/Oグラフはネットワークトラフィックを時系列でグラフィカルに表示するため、マイクロバーストを特定しやすくなります。I/Oグラフを作成するに

は、Statistics > I/O Graphの順に選択します。

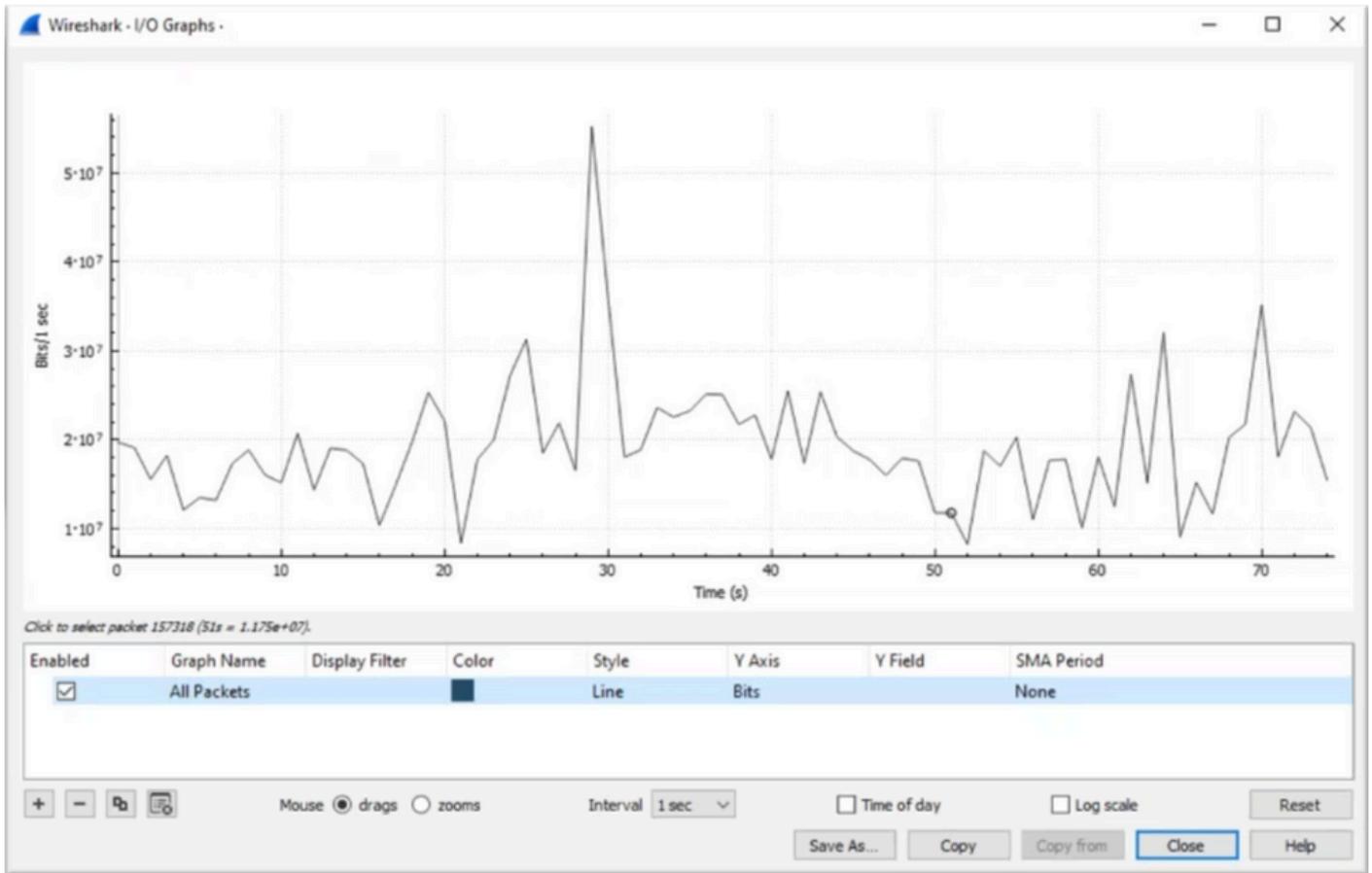
 注：この手順を示すために、1 Gbpsインターフェイスで取得したパケットキャプチャを使用しました。ただし、この手順は、高速インターフェイスでの出力廃棄のトラブルシューティングと同じです。

図 2：I/Oグラフを選択します。



次のグラフは、転送中のデータの量を示す線を示しています。グラフ内のスパイクを探します。これは、トラフィックが多い時間帯を示しています。これらのスパイクの後にトラフィックが低い期間またはトラフィックがない期間が続く場合は、マイクロバーストが発生している可能性があります。

図 3：パケットキャプチャのI/Oグラフを表示します。



すべてのパケットが選択され、表示フィルタが適用されていないことを確認することが重要です。また、Line Graphオプションを選択してY軸をBitsに設定すると、トラフィックが適切に分析されます。

図 4 : Line Graphオプションを選択して、Y軸をBitsに設定する方法を示します。



大きなパケットキャプチャを分析する場合は、対象とする特定の期間を特定することが重要です。たとえば、このシナリオでは、約30秒で大量のトラフィックが発生していることが確認できます。

I/Oグラフのスパイクのピークをクリックすると、Wiresharkはバックグラウンドでそのパケットを選択します。このシナリオでは、マイクロバーストの存在が疑われるタイムスタンプ内で動作するために、パケットキャプチャのサブセットを作成するためにパケット79280と101896が選択されています。

図 5. マイクロバーストの存在が疑われるタイムスタンプに焦点を合わせたパケットキャプチャのサブセットを作成する方法を示します。

No.	Time	Source	Destination	Protocol	Length	Info
79278	27.984062	52...	10...	Expedited Forward...	51493 → 50004	Len=521
79279	27.989775	64...	10...	Default,Default		Application Data
79280	27.995552	40...	10...	Default,Default		Application Data

Packet 79280 selected

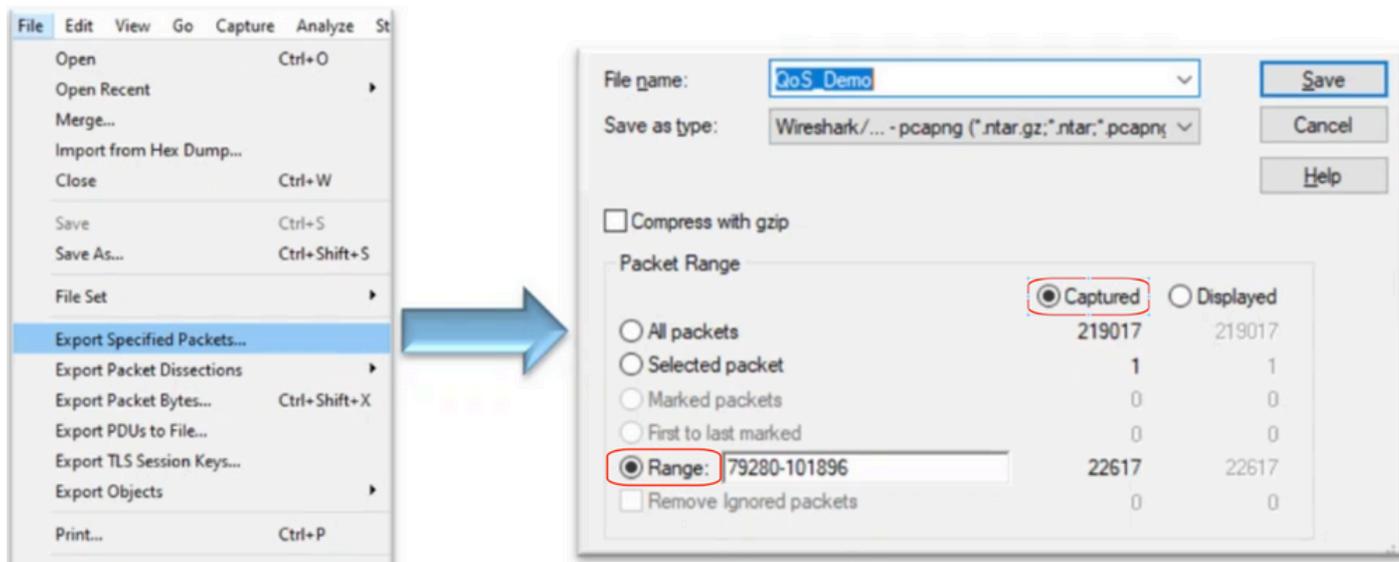
No.	Time	Source	Destination	Protocol	Length	Info
101893	33.930233	23...	10...	Default,Default		ESP (SPI=0A3E0306)
101894	33.963462	40...	10...	Default,Default		[TCP Retransmission] 443
101895	33.970760	52...	10...	Assured Forwardin...		Binding Success Response
101896	33.981855	52...	10...	Expedited Forward...	51493 → 50004	Len=521
101897	34.006120	52...	10...	Assured Forwardin...		Binding Success Response

Packet 101896 selected

選択した最初と最後のパケットを新しいファイルにエクスポートするには、RangeとCapturedの

ラジオアイコンを必ず選択してください。

図 6.パケットキャプチャのサブセットをエクスポートして保存する方法を示します。



ファイルを保存したら、ファイルを開いてI/Oグラフに戻ります。スパイクをミリ秒単位でグラフ化するには、間隔を1ミリ秒に設定します。

図 7書き出されたパケットキャプチャのサブセットのI/Oグラフを表示します。



グラフを表示する際には、トラフィックがフルミリ秒のラインレートで送信された期間を表すスパイクを特定することが重要です。この期間はトラフィックによってバッファがいっぱいになり

、出力廃棄が発生した可能性があります。たとえば、1 Gbpsインターフェイスでは、1,000,000ビット/ミリ秒に相当します。マイクロバーストの可能性を示すスパイクのピークをクリックします。これは、出力ドロップの原因となったパケットを特定するのに役立ちます。このパケットをさらに分析して、マイクロバーストの根本原因を特定し、是正措置を講じることができます。

図 8.I/Oグラフで潜在的なマイクロバーストトラフィックを特定する方法を示します。





警告：高速インターフェイスでWiresharkまたはその他のパケットキャプチャツールを使用する場合は、この制限を認識することが重要です。40Gや100Gなどの高速インターフェイスは、大量のネットワークトラフィックを生成し、パケットのキャプチャに使用されるシステムのリソースを圧倒する可能性があります。その結果、キャプチャプロセス中にパケットのドロップが発生し、キャプチャデータの正確性と完全性に影響を与える可能性があります。

---

## 代替アプローチ

キューに割り当てられたリソースを使い果たしても引き続きドロップが発生する場合は、輻輳を管理するための代替オプションを検討する必要があります。これには次のものが含まれます。

- インターフェイス速度のアップグレード。1 Gから10 G、10 Gから25 G、または40 Gに移行することで、出力帯域幅を増やし、オーバーサブスクリプション率を下げるすることができます。
- キュー単位/インターフェイス単位のバッファが大きいプラットフォームに変更します。
- アプリケーションの設定を調整して、出力廃棄の原因となるバーストのサイズを小さくしま

す。

- キューイングスケジューラを使用して、トラフィックの1つのクラスを他のクラスよりも優先します。この方法では、重要度の高いトラフィックの保護に優先順位を付けますが、重要度の低いトラフィックの廃棄は増加します。
- Weighted Random Early Discard ( WRED ; 重み付けランダム早期廃棄 ) や Weighted Tail Drop ( WTD ; 重み付けテールドロップ ) などの輻輳管理アルゴリズムを実装し、トラフィックを早期にドロップします。WREDまたはWTDのしきい値により、バーストトラフィックの早期の廃棄が引き起こされ、その結果、エンドクライアントの送信ウィンドウが自動的に短縮される可能性があります。これにより、他のバースト性の低いトラフィックが輻輳の減少に遭遇し、輻輳が高いときにさまざまなトラフィッククラスに対して最小限のバッファ割り当てが保証されます。
- データバックアップなどの既知の高帯域幅アプリケーションのソースに最も近いトラフィックをポリシングし、ネットワーク上のバーストの頻度と重大度を減らします。これにより、ネットワーク上の別の場所に存在する、送信元と宛先間の低帯域幅エリアに、バーストトラフィックと通常のネットワークトラフィックの両方を効果的に管理する機会が増えます。
- Port-channelを使用します。ただし、ハッシュにより、複数のフローが1つのメンバーに向けられる可能性があり、これが永続的なドロップにつながる可能性があることに注意することが重要です。

これらのオプションの中には、トラフィックエンジニアリングなど、より複雑な設定を必要とするものもありますが、ネットワークの輻輳や出力のドロップを緩和するという大きな利点があることに注意してください。

## 関連情報

- [『Cisco Catalyst 9000スイッチングプラットフォーム：QoSとキューイング』ホワイトペーパー](#)
- [Quality of Serviceコンフィギュレーションガイド、Cisco IOS XE 17.x](#)
- [Cisco Optics-to-Device互換性マトリクス](#)
- [サポートの革新：Cisco TACによるドキュメントの変革とセルフサービスの簡素化](#)
- [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。