

アクセスレイヤスイッチでの QoS による出力ドロップのトラブルシューティングに関するテクニカルノート

目次

[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](#)

[機能情報](#)

[トラブルシューティング方法](#)

[一般的な問題](#)

[よく寄せられる質問 \(FAQ\)](#)

[関連情報](#)

概要

このドキュメントでは、Cisco Catalyst スイッチ シリーズ 2960、3750、3750G、3750X、3560 の Quality of Service (QoS) による出力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

QoS に関する基本的な知識があることが推奨されます。

使用するコンポーネント

このドキュメントの情報は、次のプラットフォームに基づいています。Cisco Catalyst スイッチ シリーズ 2960、3750、3750G、3750X、3560。

本書の情報は、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、初期 (デフォルト) 設定の状態から起動しています。稼働中のネットワークで作業を行う場合、コマンドの影響について十分に理解したうえで作業してください。

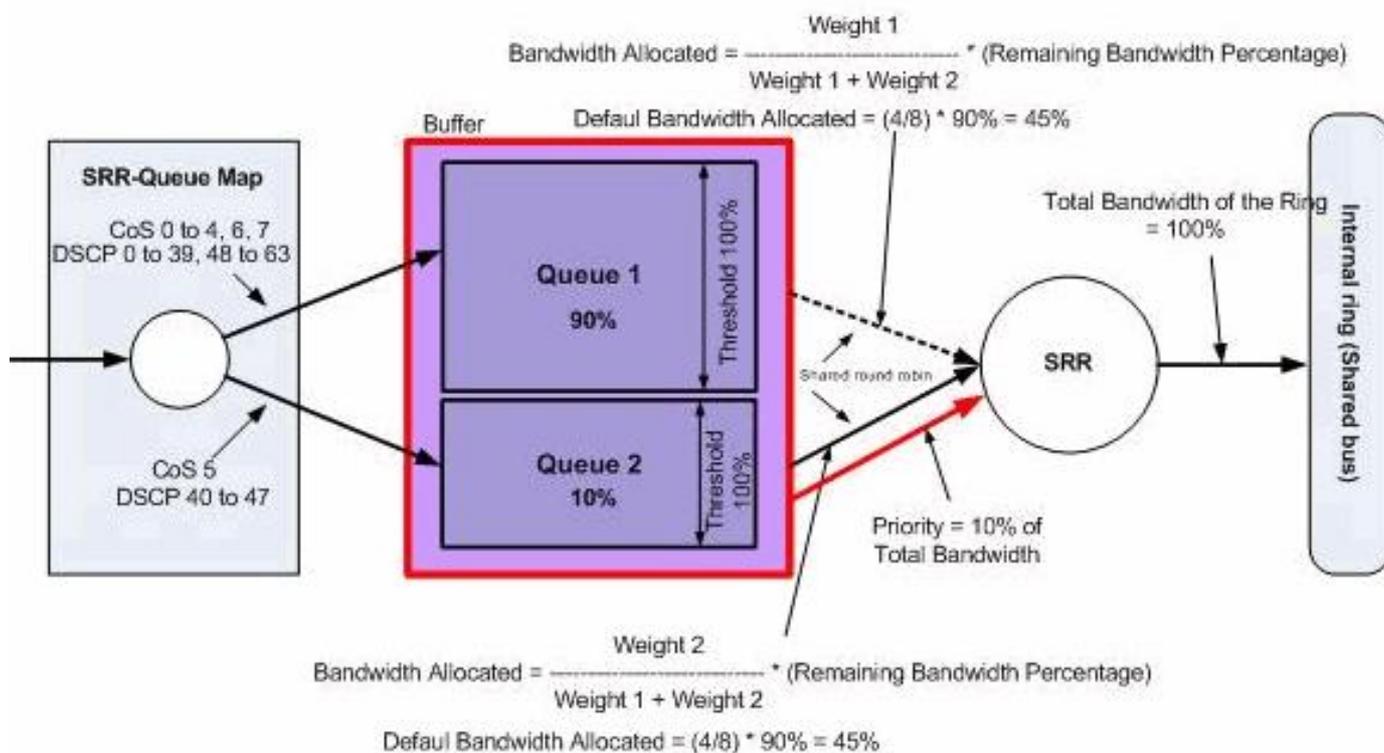
背景説明

QoS は輻輳時に、重要なデータの順位付けに使用されます。その結果、QoS を有効にすると、それほど重要でないバルク データがドロップされる場合があります。

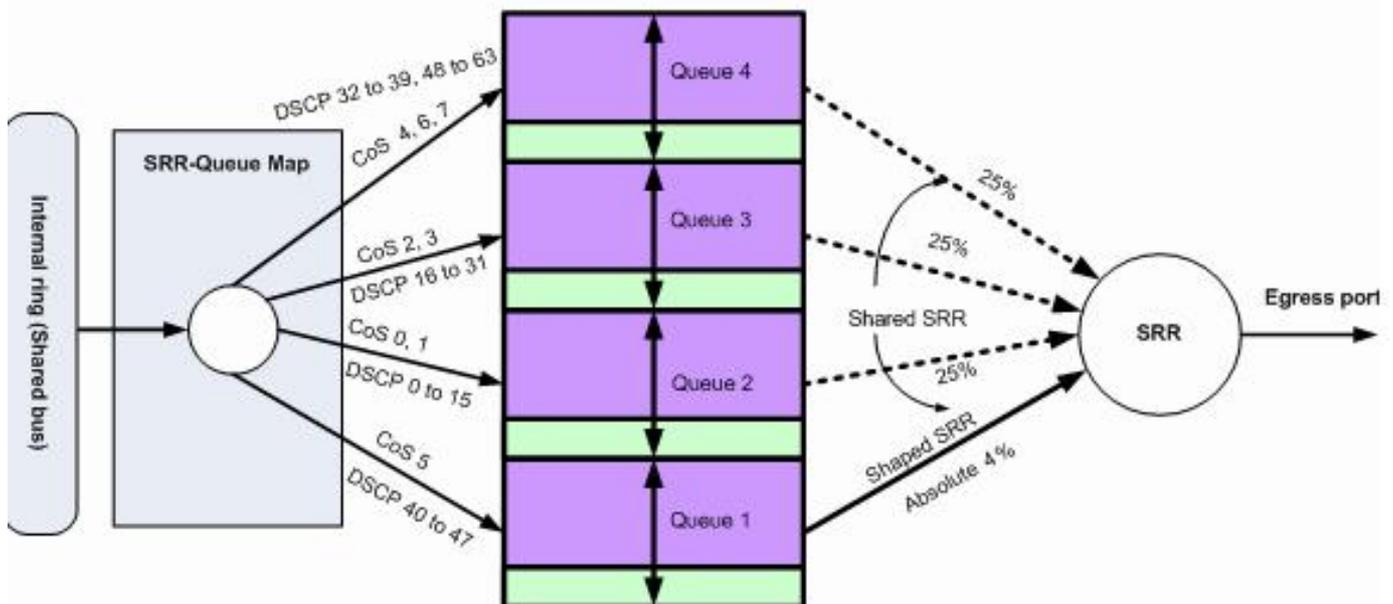
Cisco アクセスレイヤ スイッチでは、ハードウェアに QoS 機能が実装されています。このドキュメントは、ドロップが QoS によって発生したかどうかを確認するのに役立ち、ドロップを緩和するさまざまなキューイングとバッファ チューニング オプションについて説明します。

機能情報

入力デフォルト キュー



出力デフォルト キュー



トラブルシューティング方法

1. 影響を受けたアプリケーションの出力データを伝送するインターフェイス、または増加する出力ドロップが発生しているインターフェイスを特定します。インターフェイスの出力レートとインターフェイス速度を比較し、ドロップの原因がリンクの過度な使用率でないことを確認します。

```
Switch#show int gi1/0/1
!-- Some output omitted.
GigabitEthernet0/1 is up, line protocol is up (connected)
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
Full-duplex, 1000Mb/s, media type is 10/100/1000BaseTX

input flow-control is off, output flow-control is unsupported
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1089

Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 4000 bits/sec, 6 packets/sec
5 minute output rate 3009880 bits/sec, 963 packets/sec
```

2. スイッチで QoS が有効になっていることを確認します。有効になっていない場合は、出力ドロップは QoS とは関係なく、したがって、ここに示す詳細な手順は関係ありません。

```
Switch#show mls qos
QoS is enabled
QoS ip packet dscp rewrite is enabled
```

3. インターフェイスでドロップされる発信トラフィックのマーキングを特定します。

```
Switch#show mls qos int gi1/0/1 statistics

GigabitEthernet1/0/1 (All statistics are in packets)

dscp: incoming
-----

0 - 4 : 0 0 0 0 0
5 - 9 : 0 0 0 0 0
10 - 14 : 0 0 0 0 0
```

```
15 - 19 : 0 0 0 0 0
20 - 24 : 0 0 0 0 0
25 - 29 : 0 0 0 0 0
30 - 34 : 0 0 0 0 0
35 - 39 : 0 0 0 0 0
40 - 44 : 0 0 0 0 0
45 - 49 : 0 198910 0 0 0
50 - 54 : 0 0 0 0 0
55 - 59 : 0 0 0 0 0
60 - 64 : 0 0 0 0
```

dscp: outgoing

```
-----
0 - 4 : 0 0 0 0 0
5 - 9 : 0 0 0 0 0
10 - 14 : 0 0 0 0 0
15 - 19 : 0 0 0 0 0
20 - 24 : 0 0 0 0 0
25 - 29 : 0 0 0 0 0
30 - 34 : 0 0 0 0 0
35 - 39 : 0 0 0 0 0
40 - 44 : 0 0 0 0 0
45 - 49 :      0  248484      0      0      0
50 - 54 : 0 0 0 0 0
55 - 59 : 0 0 0 0 0
60 - 64 : 0 0 0 0
```

cos: incoming

```
-----
0 - 4 : 2 0 0 0 0
5 - 7 : 0 0 0
```

cos: outgoing

```
-----
0 - 4 : 0 0 0 0 0
5 - 7 : 0 0 0
```

output queues enqueued:

queue: threshold1 threshold2 threshold3

```
-----
queue 0: 248484 0 0
queue 1: 0 0 0
queue 2: 0 0 0
queue 3: 0 0 0
```

output queues dropped:

queue: **threshold1** threshold2 threshold3

```
-----
queue 0:  1089      0      0
queue 1: 0 0 0
queue 2: 0 0 0
queue 3: 0 0 0
```

Policer: Inprofile: 0 OutofProfile: 0

注: この例は、ueue 0/threshold1 でドロップされたパケットのドロップされたパックを示します。ドキュメントの他の例では、キュー番号は 1 ~ 4 です。そのため、この値は queue 1 になります。

4. どのキューとしきい値のペアがドロップされたマーキングにマッピングされるかを特定するため、スイッチで output-q マップへのマーキングを確認します。このシナリオでは、

queue1/threshold1 は、インターフェイスでドロップされる dscp 46 にマッピングされています。これは、dscp 46 のトラフィックが queue1 に送信され、このキューのバッファが不足しているか、または CPU サイクルが少ないため、ドロップされることを意味します。

```
Switch#show mls qos maps dscp-output-q
```

```
Dscp-outputq-threshold map:
```

```
d1 :d2 0 1 2 3 4 5 6 7 8 9
```

```
-----  
0 : 02-01 02-01 02-01 02-01 02-01 02-01 02-01 02-01 02-01 02-01  
1 : 02-01 02-01 02-01 02-01 02-01 02-01 03-01 03-01 03-01 03-01  
2 : 03-01 03-01 03-01 03-01 03-01 03-01 03-01 03-01 03-01 03-01  
3 : 03-01 03-01 04-01 04-01 04-01 04-01 04-01 04-01 04-01 04-01  
4 : 01-01 01-01 01-01 01-01 01-01 01-01 01-01 01-01 04-01 04-01  
5 : 04-01 04-01 04-01 04-01 04-01 04-01 04-01 04-01 04-01 04-01  
6 : 04-01 04-01 04-01 04-01
```

5. このドロップを解決するには、次の 2 つの方法があります。最初の方法は、パケットをドロップするキューのバッファとしきい値を変更することです。2 つ目の方法は、パケットをドロップするキューが他のキューよりも頻繁に処理されるようにスケジューラを設定します。

この手順では、影響を受けるキューのバッファとしきい値を変更する方法とステップ 4 で特定されたキューに関連付けられたバッファとしきい値を確認する方法を示します。注: 各キューセットには、4 つの出力キューのバッファサイズとしきい値を設定するためのオプションがあります。続いて、キューセットの任意の 1 つを任意のポートに適用できます。デフォルトでは、queue-set 2 を使用するように明示的に設定されていない限り、すべてのインターフェイスは出力キューに queue-set 1 を使用します。このシナリオでは、queue-set 1 の queue 1 には合計バッファスペースの 25% があり、しきい値 1 は 100% に設定されています。

```
Switch#show mls qos queue-set
```

```
Queueset: 1
```

```
Queue : 1 2 3 4
```

```
-----  
buffers : 25 25 25 25  
threshold1: 100 200 100 100  
threshold2: 100 200 100 100  
reserved : 50 50 50 50  
maximum : 400 400 400 400
```

```
Queueset: 2
```

```
Queue : 1 2 3 4
```

```
-----  
buffers : 25 25 25 25  
threshold1: 100 200 100 100  
threshold2: 100 200 100 100  
reserved : 50 50 50 50  
maximum : 400 400 400 400
```

6. 影響を受けるインターフェイスだけのバッファとしきい値を変更する場合は、queue-set 2 を変更し、queue-set 2 を使用するように影響を受けるインターフェイスを設定します。注: queue-set 1 も変更することができます。ただし、デフォルトではすべてのインターフェイスが queue-set 1 を使用するため、変更がすべてのインターフェイスに反映されます。この例では、queue 1 が合計バッファの 70% を受け取るように queue-set 2 を変更します。

```
Switch(config)#mls qos queue-set output 2 buffers 70 10 10 10
```

この例では、queue-set 2 と queue 1 のしきい値を変更します。必要に応じて予約されたプールからバッファを引き出すことができるように、threshold 1 と threshold 2 の両方が 3100 にマッピングされます。

```
Switch(config)#mls qos queue-set output 2 threshold 1 3100 3100 100 3200
```

7. 正しいキューとキュー セットに変更が反映されることを確認します。

```
Switch#show mls qos queue-set
Queueset: 1
Queue : 1 2 3 4
-----
buffers : 25 25 25 25
threshold1: 100 200 100 100
threshold2: 100 200 100 100
reserved : 50 50 50 50
maximum : 400 400 400 400
Queueset: 2
Queue : 1 2 3 4
-----
buffers : 70      10      10      10
threshold1: 3100    100     100     100
threshold2: 3100  100  100  100
reserved : 100     50      50      50
maximum : 3200    400     400     400
```

8. 影響を受けるインターフェイスに変更が反映されるように、このインターフェイスが queue-set 2 を使用するようにします。

```
Switch(config)#int gi1/0/1
Switch(config-if)#queue-set 2
Switch(config-if)#end
```

インターフェイスが queue-set 2 にマッピングされていることを確認します。

```
Switch#show run int gi1/0/1
interface GigabitEthernet1/0/1
switchport mode access
mls qos trust dscp
queue-set 2
end
```

インターフェイスがパケットを引き続きドロップするかどうかを確認します。

9. また、共有とシェーピングのオプションを使用して queue 1 が処理されるレートを増やすようにスケジューラを設定することもできます。この例では、queue 1 は単独で合計 CPU サイクルの 50% を受け取り、他の 3 つのキューはまとめて CPU サイクルの 50% を受け取ります。

```
Switch(config-if)#srr-queue bandwidth share 1 75 25 5
Switch(config-if)#srr-queue bandwidth shape 2 0 0 0
```

インターフェイスがパケットを引き続きドロップするかどうかを確認します。

10. このインターフェイス上でプライオリティ キューを有効にします。この操作は、プライオリティ キューのすべてのトラフィックが他のキューより先に処理されることを確実にします。注: 他のキューを処理する前に、プライオリティ キューは空になるまで処理されます。デフォルトでは、スイッチ 2960、3560、3750 では、queue 1 がプライオリティ キューです。

```
Switch(config)#int gi1/0/1
Switch(config-if)#priority-queue out
Switch(config-if)#end
```

インターフェイスでドロップされるパケットのマーキングを queue 1 (プライオリティ キュー) に送るようにマッピングできます。この操作は、このマーキングがあるトラフィックが必ず先に処理されることを確実にします。

```
Switch(config)#mls qos srr-queue output dscp-map queue 1 threshold 1
```

一般的な問題

次に、いくつかの一般的な問題を示します。

- QoS を有効にした後、インターフェイスで出力がドロップされます。
- ボイスコールが途切れ途切れになります。
- 追加された遅延によって、ビデオトラフィックが最適な状態ではなくなります。
- 接続がリセットされます。

FAQ

Q: キュー セットはいつ変更しますか、また共有とシェーピングはいつ使用しますか。

A: この決定は、ドロップの性質によって異なります。ドロップが断続的に増加する場合、この問題はバーストトラフィックが原因である可能性があります。その一方でドロップが常に一定のレートで増加する場合、パケットをドロップするキューは送信できるデータよりも多くのデータを受信している可能性があります。

断続的なドロップの場合、キューは一時的なバーストに対応できる大きなバッファを必要とします。このソリューションを実行するには、キュー セットを変更して影響を受けるキューに、より多くのバッファを割り当てて、しきい値も増やす必要があります。

連続的なドロップの場合は、影響を受けるキューが処理される頻度を増やし、CPU サイクルごとにキューから取り除くパケットの量を増やすようにスケジューラを設定します。このソリューションを実行するには、出力キューで共有またはシェーピングを設定する必要があります。

Q: 共有モードとシェーピング済みモードの違いは何ですか。

A: シェーピング済みモードでは、出力キューには帯域幅の比率が保証され、出力キューはその総量にレート制限されています。シェーピングされるトラフィックは、リンクがアイドル状態であっても、割り当てられた帯域幅以上は使用しません。シェーピング済みモードにより、時間の流れの中で均等なトラフィックのフローが実現され、バーストトラフィックの浮き沈みが緩和されます。シェーピングの場合は、各重みの絶対値を使用して、キューに使用可能な帯域幅が計算されます。

`srr-queue bandwidth shape weight1 weight2 weight3 weight4`

このキューのシェーピング帯域幅は、インバース比率 ($1/\text{weight}$) によって制御されます。つまり、queue1 は合計帯域幅の reserved $1/\text{weight1}$ などになります。重み 0 を設定した場合は、対応するキューが共有モードで動作します。 `srr-queue bandwidth shape configuration` コマンド `srr-queue bandwidth share interface configuration` コマンドでキューに指定された重み付けが反映されます。

共有モードでは、設定された重み付けに基づいてキュー間で帯域幅が共有されます。帯域幅はこのレベルで保証されますが、このレベルに制限されるものではありません。たとえば、特定のキューが空であり、リンクを共有する必要がない場合、残りのキューは未使用の帯域幅を使用して、共有できます。

`srr-queue bandwidth share weight1 weight2 weight3 weight4`

queue1 には最低 $\text{weight1} / (\text{weight1} + \text{weight2} + \text{weight3} + \text{weight4})$ パーセントの帯域幅が保証されますが、必要に応じて他のシェーピングされていないキューの帯域幅を消費することもできます。

関連情報

- [テクニカル サポートとドキュメント – Cisco Systems](#)