

# 액세스 레이어 스위치의 QoS 문제 해결 TechNote로 인한 출력 삭제

## 목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 정보](#)

[기능 정보](#)

[문제 해결 방법론](#)

[일반적인 문제](#)

[자주 묻는 질문](#)

[관련 정보](#)

## 소개

이 문서에서는 Cisco Catalyst Switches Series 2960, 3750, 3750G, 3750X, 3560에서 QoS(quality of service)로 인한 출력 삭제를 해결하는 방법에 대해 설명합니다.

## 사전 요구 사항

### 요구 사항

QoS에 대한 기본적인 지식이 있는 것이 좋습니다.

### 사용되는 구성 요소

이 문서의 정보는 다음 플랫폼을 기반으로 합니다. Cisco Catalyst 스위치 시리즈 2960, 3750, 3750G, 3750X, 3560.

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우, 모든 명령어의 잠재적인 영향을 미리 숙지하시기 바랍니다.

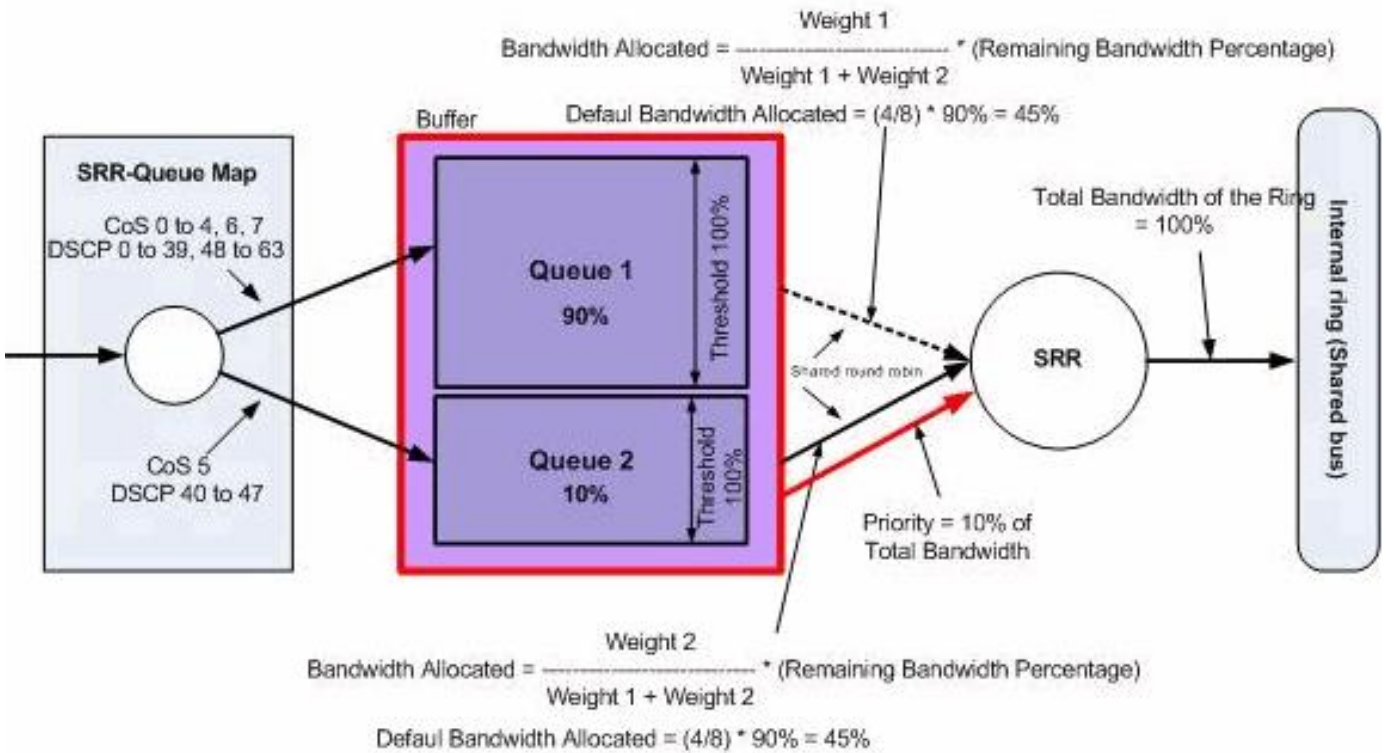
## 배경 정보

QoS는 혼잡 시 더 중요한 데이터의 우선 순위를 지정하는 데 사용됩니다. 따라서 QoS를 활성화하면 덜 중요한 대량 데이터가 삭제될 수 있습니다.

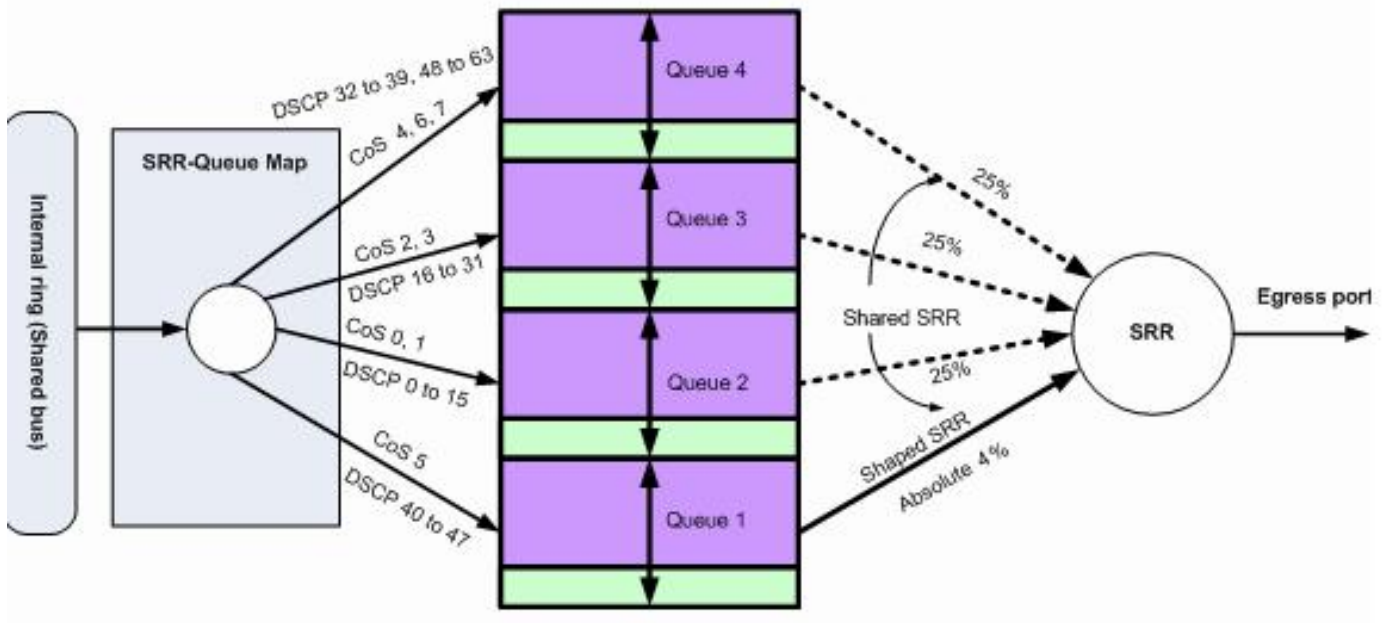
Cisco 액세스 레이어 스위치는 하드웨어에 QoS 기능을 구현합니다. 이 문서에서는 드롭이 QoS에 의해 발생하는지 확인하고 이를 완화하기 위한 다양한 대기열 및 버퍼 조정 옵션에 대해 설명합니다.

## 기능 정보

### 인그레스 기본 대기열



### 이그레스 기본 대기열



## 문제 해결 방법론

- 영향을 받는 애플리케이션의 발신 데이터를 전달하는 인터페이스 또는 증가되는 출력 삭제를 경험으로 식별합니다. 인터페이스 출력 속도와 인터페이스 속도를 비교하고 링크의 과다 사용으로 인해 드롭이 발생하지 않는지 확인합니다.

```
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
```

- 스위치에서 QoS가 활성화되어 있는지 확인합니다. 이 옵션이 활성화되지 않은 경우 출력 삭제는 QoS와 관련이 없으므로 여기에 언급된 추가 단계는 관련이 없습니다.

```
Switch#show mls qos
```

```
QoS is enabled
```

```
QoS ip packet dscp rewrite is enabled
```

- 인터페이스에서 삭제된 발신 트래픽의 표시를 식별합니다.

```
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

**참고:** 이 예에서는 패킷을 삭제하는 큐 0/threshold1의 삭제된 팩을 보여 줍니다. 문서의 다른 예제에서는 큐 번호 매기기를 1 - 4로 지정합니다. 따라서 이 값은 queue 1이 됩니다.

- 어떤 queue-threshold 쌍이 삭제된 마킹에 매핑되는지 확인하려면 스위치에서 output-q 맵에 대한 마킹을 확인합니다. 이 시나리오에서는 queue1/threshold1이 dscp 46에 매핑되며, 이 값은 인터페이스에서 삭제됩니다. 즉, dscp 46 트래픽이 queue1에 전송되고 해당 큐의 버퍼 또는 CPU 주기가 부족하여 삭제됩니다.

Switch#show mls qos maps dscp-output-q

```
Dscp-outputq-threshold map:
dl :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 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. 이러한 삭제를 해결하는 방법에는 두 가지가 있습니다. 첫 번째 방법은 패킷을 삭제하는 큐의 버퍼 및 임계값 값을 변경하는 것입니다. 두 번째 방법은 패킷을 삭제하는 큐가 나머지 대기열보다 더 자주 서비스되도록 스케줄러를 구성하는 것입니다.

이 단계는 영향을 받는 대기열의 버퍼와 임계값을 변경하고 4단계에서 식별된 대기열과 관련된 버퍼 및 임계값 값을 확인하는 방법을 보여줍니다. **참고:** 각 대기열 집합에는 4개의 이그레스 대기열에 대한 버퍼 크기 및 임계값을 구성하는 옵션이 있습니다. 그런 다음 대기열 세트 중 하나를 포트에 적용할 수 있습니다. 기본적으로 모든 인터페이스는 queue-set 2를 사용하도록 명시적으로 구성되지 않은 경우 출력 대기열에 대해 queue-set 1을 사용합니다. 이 시나리오에서는 queue-set 1의 queue 1이 총 버퍼 공간의 25%를 가지며 임계값 1은 100%로 설정됩니다.

```
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. 영향을 받는 인터페이스에 대한 버퍼 및 임계값 값만 변경하려면 queue-set 2를 변경하고 영향을 받는 인터페이스가 queue-set 2를 사용하도록 구성합니다. **참고:** queue-set 1도 변경할 수 있습니다. 그러나 모든 인터페이스는 기본적으로 queue-set 1을 사용하므로 변경 사항이 모든 인터페이스에 반영됩니다. 이 예에서는 큐 1이 총 버퍼의 70%를 수신하도록 queue-set 2가 변경됩니다.

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

이 예에서는 queue-set 2 및 queue 1 임계값이 변경됩니다. 임계값 1과 임계값 2는 모두 3100에 매핑되므로 필요한 경우 예약된 풀에서 버퍼를 가져올 수 있습니다.

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

7. 변경 사항이 올바른 대기열 및 대기열 세트 아래에 반영되는지 확인합니다.

```
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. 영향을 받는 인터페이스에서 queue-set 2를 사용하여 변경 사항이 이 인터페이스에 적용되도록 합니다.

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

인터페이스가 queue-set 2에 매핑되었는지 확인합니다.

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

인터페이스가 패킷을 계속 삭제하는지 확인합니다.

9. 또한 공유 및 모양 옵션을 사용하여 대기열 1이 서비스되는 속도를 높이도록 스케줄러를 구성할 수 있습니다. 이 예에서 대기열 1만 총 CPU 사이클의 50%를 수신하고 나머지 3개 대기열은 CPU 사이클의 50%를 전체적으로 수신합니다.

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

인터페이스가 패킷을 계속 삭제하는지 확인합니다.

10. 이 인터페이스에서 우선순위 큐를 활성화합니다. 이 작업을 수행하면 우선 순위 대기열의 모든 트래픽이 다른 대기열보다 먼저 처리됩니다. **참고:** 우선 순위 대기열은 다른 대기열이 서비스되기 전에 비어 있을 때까지 처리됩니다. 2960/3560/3750 스위치에서 기본적으로 대기열 1은 우선순위 대기열입니다.

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

인터페이스에서 삭제된 패킷의 표시를 매핑하여 대기열 1(우선순위 대기열)으로 이동할 수 있습니다. 이 작업은 이 표시가 있는 트래픽이 항상 다른 어떤 항목보다 먼저 처리되도록 합니다.

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

## 일반적인 문제

다음은 몇 가지 일반적인 문제입니다.

- QoS가 활성화된 후 인터페이스에서 출력이 삭제됩니다.
- 음성 통화 고르지 않음
- 지연이 추가되면 최적 상태가 아닌 비디오 트래픽이 발생합니다.
- 연결이 재설정됩니다.

## 자주 묻는 질문

Q: 큐 세트는 언제, 공유/셰이핑을 언제 변경합니까?

**A:**그 결정은 물방울의 성격에 따라 달라진다.가 간헐적으로 증가하면 이 문제는 버스트 트래픽 때문일 가능성이 높습니다.반대로, 삭제가 지속적으로 일정한 속도로 증가하면, 패킷을 삭제하는 대기열이 전송 가능한 것보다 더 많은 데이터를 수신합니다.

간헐적인 삭제의 경우 대기열에는 간헐적인 버스트를 수용할 수 있는 큰 버퍼가 있어야 합니다.이 솔루션을 구현하려면 큐 세트를 변경하고 영향을 받는 대기열에 더 많은 버퍼를 할당하고 임계값도 증가시켜야 합니다.

연속 삭제를 위해서는 영향을 받는 대기열을 더 자주 서비스하고 CPU 주기당 대기열에서 더 많은 패킷을 제외하도록 스케줄러를 구성해야 합니다. 이 솔루션을 구현하려면 이그레스 큐에서 공유/셰이핑을 구성해야 합니다.

#### **Q:공유 모드와 모양 모드의 차이점은 무엇입니까?**

**A:**셰이핑 모드에서는 이그레스 큐가 대역폭의 백분율을 보장하며, 그 양만큼 속도가 제한됩니다.셰이핑된 트래픽은 링크가 유휴 상태일 경우에도 할당된 대역폭보다 많은 대역폭을 사용하지 않습니다. Shaped(모양) 모드는 시간에 따라 더 많은 트래픽 플로우를 제공하고 버스트 트래픽의 최고와 계곡을 줄입니다.셰이핑을 사용하면 각 가중치의 절대 값을 사용하여 대기열에 사용할 수 있는 대역폭을 계산합니다.

#### **srr 대기열 대역폭 모양 무게1 무게2 무게3 무게4**

역역 비율(1/가중치)은 이 큐의 셰이핑 대역폭을 제어합니다. 즉, queue1은 총 대역폭의 1/weight1%에 예약되어 있습니다.가중치 0을 구성하면 해당 대기열이 공유 모드에서 작동합니다. **.srr-queue 대역폭 모양 명령**으로 지정된 가중치는 무시되며, 대기열에 대한 **srr-queue 대역폭 공유 인터페이스** 컨피그레이션 명령으로 지정된 가중치가 적용됩니다.

공유 모드에서는 큐가 구성된 가중치를 기반으로 대역폭을 공유합니다.대역폭은 이 수준에서 보장되지만 이에 국한되지 않습니다. 예를 들어, 대기열이 비어 있고 더 이상 링크 공유가 필요하지 않은 경우, 나머지 대기열은 사용되지 않은 대역폭으로 확장되어 이들 간에 공유할 수 있습니다.

#### **srr 대기열 대역폭 공유 무게1 무게2 무게3 무게4**

queue1은 대역폭의 최소 가중치1/(weight1 + weight2 + weight3 + weight4)%를 보장하지만 필요한 경우 다른 셰이프가 아닌 대기열의 대역폭까지 확장할 수 있습니다.

## **관련 정보**

- [기술 지원 및 문서 - Cisco Systems](#)