

# Productreducties vanwege QoS op Access Layer-switches - TechNotes voor probleemoplossing

## Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Achtergrondinformatie](#)

[Functieinformatie](#)

[Methode voor probleemoplossing](#)

[Vaak voorkomende problemen](#)

[Veelgestelde vragen](#)

[Gerelateerde informatie](#)

## Inleiding

Dit document beschrijft hoe u problemen kunt oplossen bij uitvoerdruppels vanwege de kwaliteit van de service (QoS) op Cisco Catalyst-switches Series 2960, 3750, 3750G, 3750X, 3560.

## Voorwaarden

### Vereisten

Cisco raadt u aan basiskennis van QoS te hebben.

### Gebruikte componenten

De informatie in dit document is gebaseerd op de volgende platforms: Cisco Catalyst-switches Series 2960, 3750, 3750G, 3750X, 3560.

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u de potentiële impact van elke opdracht begrijpen.

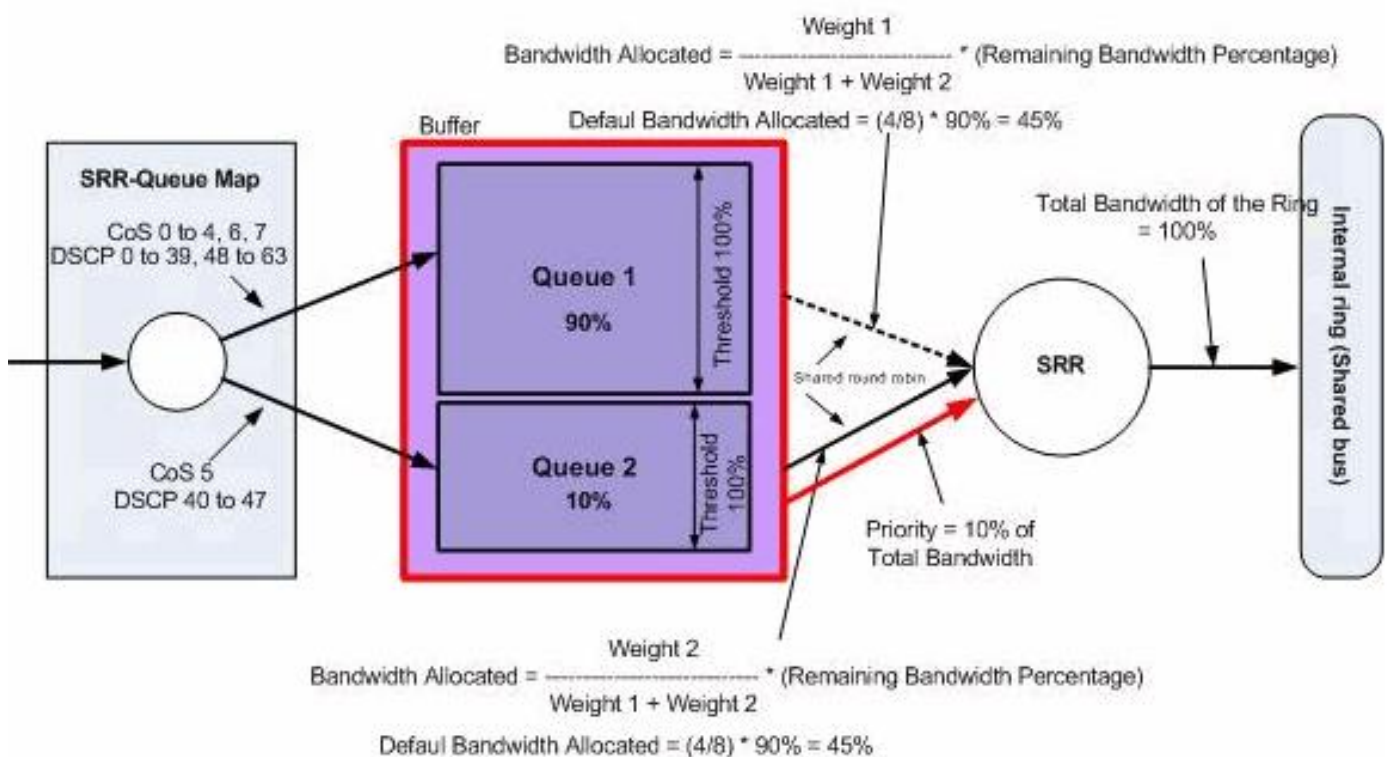
# Achtergrondinformatie

QoS wordt gebruikt om belangrijker gegevens tijdens congestie voorrang te geven. Als resultaat hiervan, nadat QoS is ingeschakeld, kunnen minder belangrijke bulkgegevens druppels ervaren.

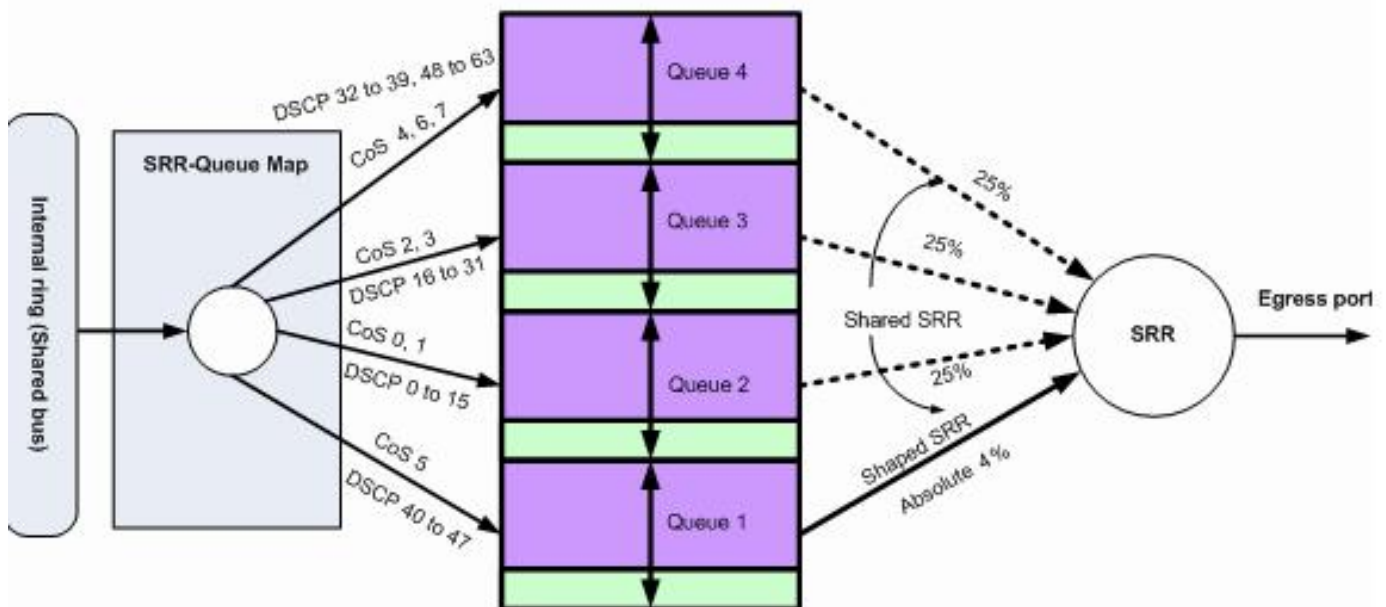
Cisco-toegangslaagswitches implementeren QoS-functies in hardware. Dit document helpt u vast te stellen of de vallen door QoS worden veroorzaakt en beschrijft verschillende opties voor het in de wachtrij en buffer afstemmen om deze te verzachten.

## Functieinformatie

### Standaardwachtrij voor invoeren



### Standaard wachtrij voor uitzetten



## Methode voor probleemoplossing

1. Identificeer de interfaces die uitgaande gegevens voor de getroffen toepassing dragen of die ervaringsoutput die toename vermindert. Vergelijk de interface-uitvoersnelheid en de interfacesnelheid en zorg ervoor dat de druppels niet te veel gebruikt worden door de link.

```
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. Zorg ervoor dat QoS op de schakelaar is ingeschakeld. Als deze niet ingeschakeld is, zijn outputdruppels niet gerelateerd aan QoS en zijn dus verdere stappen die hier genoemd worden irrelevant.

```
Switch#show mls qos
```

```
QoS is enabled
```

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

3. Identificeer de markering van het uitgaande verkeer die op de interface wordt gedropt.

```
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

Opmerking: Dit voorbeeld toont geworpen pakjes op rij 0/drempelwaarde 1 druppelpakketten. In andere voorbeelden in het document is de nummering in de rij 1 - 4; derhalve zal deze waarde in wachtrij 1 staan .

4. Controleer de markering op uitvoer-q kaart op de schakelaar om te bepalen welke de wachtrijdrempel paar kaarten op de markering aanbrengt. In dit scenario wordt wachtrij

1/threshold1 in kaart gebracht op dscp 46, dat op de interface is gevallen. Dit betekent dat DSCP 46-verkeer naar wachtrij 1 wordt verzonden en wordt gedropt omdat de wachtrij onvoldoende buffer of minder CPU-cycli heeft.

```
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 02-01
1 : 02-01 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 03-01
3 : 03-01 03-01 04-01 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 04-01
6 : 04-01 04-01 04-01 04-01
```

- Er zijn twee methoden om deze druppels op te lossen. De eerste methode is om de buffer en de drempelwaarden voor de rij te veranderen die pakketten laat vallen. De tweede methode is om de planner te configureren zodat de rij die pakketten laat vallen vaker wordt onderhouden dan de rest van de wachtrijen.

Deze stappen tonen hoe u de buffer en de drempel voor de getroffen wachtrijen kunt veranderen en controleert de buffer en de drempelwaarden verbonden aan de rij die in stap 4 zijn geïdentificeerd. Opmerking: Elke wachtrij heeft de optie om de buffergrootte en de drempelwaarde voor de vier noodoplossingen te configureren. Vervolgens kunt u een van de wachtrijen toepassen op de poorten. Standaard gebruiken alle interfaces in de wachtrij-set 1 voor uitvoerwachtrijen, tenzij ze expliciet zijn ingesteld op gebruik van in de wachtrij 2. In dit scenario heeft wachtrij 1 in wachtrij 1 25% van de totale bufferruimte en drempel 1 is ingesteld op 100%

```
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
```

- Als u de buffer en de drempelwaarden voor slechts de getroffen interface wilt veranderen, verander de rij-set 2 en stel de getroffen interface in om wachtrij-set 2 te gebruiken. Opmerking: U kunt in de wachtrij 1 ook wijzigen; echter, aangezien alle interfaces standaard in de wachtrij 1 gebruiken, wordt de verandering op alle interfaces weergegeven. In dit voorbeeld wordt de wachtrij-set 2 veranderd zodat wachtrij 1 70% van de totale buffer ontvangt.

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

In dit voorbeeld worden de wachtrijen 2 en wachtrij 1 drempels veranderd. Zowel drempel 1 als drempel 2 worden in kaart gebracht tot 3100 zodat zij desgewenst de buffer uit de gereserveerde pool kunnen trekken.

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

7. Controleer dat de wijzigingen in de juiste wachtrij en in de wachtrij worden weergegeven.

```
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. Maak de beïnvloed interface gebruik in de wachtrij 2 zodat de wijzigingen op deze interface van kracht worden.

```
Switch(config)#int gi1/0/1
```

```
Switch(config-if)#queue-set 2
```

```
Switch(config-if)#end
```

Controleer dat de interface in kaart is gebracht in de wachtrij 2.

```
Switch#show run int gi1/0/1
```

```
interface GigabitEthernet1/0/1
```

```
switchport mode access
```

```
mls qos trust dscp
```

```
queue-set 2
```

```
end
```

Controleer of de interface pakketten blijft neerzetten.

9. U kunt ook de planner configureren om het tarief te verhogen waarbij wachtrij 1 wordt bediend met de opties aandeel en vorm. In dit voorbeeld wordt in wachtrij 1 alleen 50% van de totale CPU-cycli ontvangen en in de andere drie wachtrijen tezamen 50% van de CPU-cycli ontvangen.

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

```
Switch(config-if)#srr-queue bandwidth shape 2 0 0 0
```

Controleer of de interface pakketten blijft neerzetten.

10. Prioritaire wachtrij op deze interface inschakelen. Deze actie waarborgt dat al het verkeer in de prioriteitswachtrij wordt verwerkt vóór een andere wachtrij. Opmerking: De prioriteitswachtrij wordt onderhouden totdat de wachtrijen leeg zijn voordat de overige wachtrijen worden onderhouden. Standaard op 2960/3560/3750 switches is wachtrij 1 de prioriteitswachtrij.

```
Switch(config)#int gi1/0/1
```

```
Switch(config-if)#priority-queue out
```

```
Switch(config-if)#end
```

Markeren van het pakje dat op de interface is gevallen kan worden aangepast zodat het in de wachtrij 1 (prioriteitswachtrij) wordt geplaatst. Met deze actie wordt ervoor gezorgd dat het verkeer met deze markering altijd vóór alles wordt verwerkt.

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

## Vaak voorkomende problemen

Hier zijn een paar voorkomende problemen:

- Uitvoer druppels op interfaces nadat QoS is ingeschakeld.
- Choppie spraakoproepen.
- Toegevoegde vertraging veroorzaakt suboptimaal videoverkeer.
- Verbinding opnieuw ingesteld.

## Veelgestelde vragen

### V: Wanneer verander ik de wachtrij en wanneer gebruik ik delen/vormgeven?

**A:** De beslissing hangt af van de aard van de druppels. Als de druppels met tussenpozen worden verhoogd is dit probleem waarschijnlijk het gevolg van een overvloedig verkeer. In tegendeel, als de toename continu tegen een constant tempo daalt, ontvangt de rij die de pakketten laat vallen waarschijnlijk meer gegevens dan het uit kan sturen.

Voor intermitterende druppels moet de rij een grote buffer hebben die incidentele brandwonden kan verwerken. Om deze oplossing te implementeren moet u de wachtrijset wijzigen en meer buffer aan de getroffen wachtrij toewijzen en ook de drempelwaarden verhogen.

Voor voortdurende druppels moet u de planner configureren om de getroffen wachtrij vaker te bedienen en per CPU-programma meer pakketten uit de wachtrij te halen. Om deze oplossing te implementeren moet u het delen/vormgeven aan de stress wachtrijen bevestigen.

### V: Wat is het verschil tussen de gedeelde modus en de gevormde modus?

**A:** In de vormige modus zijn de stress wachtrijen gegarandeerd als percentage van de bandbreedte en ze zijn aan de snelheid gebonden. Vormd verkeer gebruikt niet meer dan de toegewezen bandbreedte zelfs als de verbinding onklaar is. De vormmodus zorgt voor een gelijkmatiger verkeersstroom in de loop der tijd en vermindert de pieken en dalen van het stoffige verkeer. Met het vormgeven wordt de absolute waarde van elk gewicht gebruikt om de bandbreedte die beschikbaar is voor de wachtrijen te berekenen.

### **Rr-wachtrij bandbreedte** *gewicht1 gewicht2 gewicht3 gewicht4*

De omgekeerde verhouding ( $1/\text{gewicht}$ ) bepaalt de vormende bandbreedte voor deze rij. Met andere woorden, wachtrij1 is gereserveerd voor  $1/\text{gewichtsprocent}$  van de totale bandbreedte enzovoort. Als u een gewicht van 0 configureren werkt de corresponderende rij in gedeelde modus. Het gewicht dat met de opdracht de **vormbandbreedte in de rij wordt** gespecificeerd wordt genegeerd, en de gewichten die met de opdracht **van de bandbreedte van de srr-wachtrij worden** gespecificeerd, **delen de** configuratie van de **interface** voor een wachtrij.

In gedeelde modus delen de wachtrijen de bandbreedte tussen de wachtrijen op basis van de ingestelde gewichten. De bandbreedte wordt op dit niveau gegarandeerd maar niet beperkt tot het niveau. Als een rij bijvoorbeeld leeg is en niet langer een deel van de link nodig heeft, kunnen de resterende wachtrijen in de ongebruikte bandbreedte uitbreiden en het onder hen delen.

### **bandbreedte-verdeling in rij (SRR)** *gewicht1 gewicht2 gewicht3 gewicht4*

wachtrij1 is gegarandeerd een minimum aan  $\text{gewicht1}/(\text{gewicht1} + \text{gewicht2} + \text{gewicht3} + \text{gewicht4})$  procent van de bandbreedte maar kan indien nodig ook tot de bandbreedte van andere niet-vormige rijen opeten.

## Gerelateerde informatie

- [Technische ondersteuning en documentatie – Cisco Systems](#)