QoS O-planning bij Catalyst 6500/6000 Series Switches die Cisco IOS-systeemsoftware uitvoeren

Inhoud

Inleiding

Het gebruik van uitvoerschema's zorgt ervoor dat het belangrijke verkeer niet wordt laten vallen in het geval van zware overabonnement. In dit document worden alle technieken en algoritmen besproken die bij het plannen van de uitvoer betrokken zijn, op de Catalyst 6500/6000 switch. Dit document legt ook uit hoe u de bediening van uitvoerschema's kunt configureren en controleren op Catalyst 6500/6000 dat Cisco IOS®-software draait.

Raadpleeg QoS-uitvoerplanning voor Catalyst 6500/6000 Series Switches die CatOS-systeemsoftware uitvoeren voor meer informatie over Weighted Random Early Detection (WRED), Weighted round robin (WRR) en tail drop.

Voorwaarden

Vereisten

Er zijn geen specifieke vereisten van toepassing op dit document.

Gebruikte componenten

Dit document is niet beperkt tot specifieke software- en hardware-versies.

Conventies

Raadpleeg Cisco Technical Tips Conventions (Conventies voor technische tips van Cisco) voor meer informatie over documentconventies.

Achtergrondinformatie

Stapels uitvoerwachtrij

Uitvoerdruppels worden veroorzaakt door een verstopte interface. Een gemeenschappelijke oorzaak van dit kan verkeer van een hoge bandbreedte verbinding zijn die naar een lagere bandbreedte verbinding wordt geschakeld, of verkeer van meerdere inkomende verbindingen die naar één enkele uitgaande verbinding wordt geschakeld.

Bijvoorbeeld, als een grote hoeveelheid bursty verkeer op een gigabit interface in komt en uitgeschakeld wordt naar een 100 Mbps interface, kan dit uitvoerdruppels aan toename op de 100 Mbps interface veroorzaken. Dit komt doordat de uitvoerwachtrij op die interface wordt overweldigd door het overtollige verkeer vanwege de snelheidswanverhouding tussen de inkomende en uitgaande bandbreedte. De verkeerssnelheid op de uitgaande interface kan niet alle pakketten accepteren die moeten worden verzonden.

Om het probleem op te lossen, is de beste oplossing om de lijnsnelheid te verhogen. Er zijn echter manieren om output druppels te voorkomen, te verminderen of te controleren als u de lijnsnelheid niet wilt verhogen. U kunt een daling van de uitvoer alleen voorkomen als de daling van de uitvoer een gevolg is van korte gegevensuitbarstingen. Als de afvoerdruppels worden veroorzaakt door een constante snelle stroom, kunt u de druppels niet voorkomen. Maar je kunt ze wel controleren.

Output Queueing-mogelijkheid voor verschillende lijnkaarten op Catalyst 6500/6000

Als u onzeker bent over het wachtrijen vermogen van een haven, **geef** de **tonen** een **wachtrij interface {gigabitethernet uit | fastethernet}** *mod/port* commando. Dit wordt weergegeven in de eerste uitvoerlijnen van een opdracht **voor** een **showwachtrij**. De poort is op een Supervisor Engine 1A lijnkaart:

```
cosmos#show queueing interface gigabitethernet 1/1
Interface GigabitEthernet1/1 queueing strategy:  Weighted Round-Robin

  QoS is disabled globally
  Trust state: trust DSCP
  Default COS is 0
  Transmit group-buffers feature is enabled
  Transmit queues [type = 1p2q2t]:
    Queue Id    Scheduling  Num of thresholds
```

```
    ----------------------------------------
       1         WRR low             2
       2         WRR high            2
       3         Priority            1
!--- Output suppressed.
```

De output toont dat deze poort een output een wachtrij type heeft dat bekend staat als 1p2q2t.

Een andere manier om het type van de rij beschikbaar op een specifieke haven te zien is de opdracht van de **show interfacemogelijkheden** uit te geven:

```
la-orion#show interface gigabitethernet 6/2 capabilities
GigabitEthernet6/2
  Model:                 WS-SUP720-BASE
  Type:                  No GBIC
  Speed:                 1000
  Duplex:                full
  Trunk encap. type:     802.1Q,ISL
  Trunk mode:            on,off,desirable,nonegotiate
  Channel:               yes
  Broadcast suppression: percentage(0-100)
  Flowcontrol:           rx-(off,on,desired),tx-(off,on,desired)
  Membership:            static
  Fast Start:            yes
  QOS scheduling:        rx-(1p1q4t), tx-(1p2q2t)
  CoS rewrite:           yes
  ToS rewrite:           yes
  Inline power:          no
  SPAN:                  source/destination
  UDLD                   yes
  Link Debounce:         yes
  Link Debounce Time:    yes
  Ports on ASIC:         1-2
```

Begrijp de Wachtrij-capaciteit van een poort

Er zijn verschillende soorten wachtrijen beschikbaar voor Catalyst 6500/6000 switches. In deze tabel wordt de notatie van de QoS-architectuur van de poort uitgelegd:

Verzenden (TX)/ontvangen (RX)	Wachtrij melding	Aantal rijen	Prioritaire wachtrij	Aantal WRR-wachtrijen	Aantal en type drempel voor WRR-wachtrijen
TX	2q2t	2	—	2	2 Configureerbare staartdruppels
TX	1p2q2t	3	1	2	2 configureerbare WRED
TX	1p3q1t	4	1	3	1 configureerbare WRED
TX	1p2q1t	3	1	2	1 configureer

					bare WRED
RV	1q4t	1	—	—	4 Configureer bare staart
RV	1p1q4t	2	1	1	4 Configureer bare staart
RV	1p1q0t	2	1	1	Niet configureer baar
RV	1p1q8t	2	1	1	8 configureer bare WRED
TX	1p3q8t	4	1	3	8 Configureer bare WRED of tail-drop
TX	1p7q8t	8	1	7	8 Configureer bare WRED of tail-drop
RV	1q2t	1	—	—	1 configureer bare staart - daling = 1 niet-configureer baar
RV	1q8t	1	—	—	8 Configureer bare staartdrupp els
RV	2q8t	2	—	2	8 Configureer bare staartdrupp els

De volgende tabel toont een aantal modules en wachtrijtypen in de RX- en TX-kanten van de interface of poort. Als uw module hier niet vermeld is, gebruik de opdracht **Show interface mogelijkheden** om de beschikbare rijcapaciteit te bepalen. De opdracht **interface-**functies **tonen** wordt beschreven in de uitvoercapaciteit van verschillende lijnkaarten op Catalyst 6500/6000 sectie.

Module	**RX-wachtrijen**	**TX-wachtrijen**
WS-X6K-S2-PFC2	1p1q4t	1p2q2t
WS-X6K-SUP1A-2GE	1p1q4t	1p2q2t
WS-X6K-SUP1-2 GE	1q4t	2q2t

WS-X6501-10GEX4-software	1p1q8t	1p2q1t
WS-X6502-10 GE switch	1p1q8t	1p2q1t
WS-X6516-GBIC	1p1q4t	1p2q2t
WS-X6516 GE-TX switch	1p1q4t	1p2q2t
WS-X6416-GBIC	1p1q4t	1p2q2t
WS-X6416 GE-M MT	1p1q4t	1p2q2t
WS-X6316 GE-TX switch	1p1q4t	1p2q2t
WS-X6408A-GBIC	1p1q4t	1p2q2t
WS-X6408-GBIC	1q4t	2q2t
WS-X6524-100 FX-M switch	1p1q0t	1p3q1t
WS-X6324-100FX-SM	1q4t	2q2t
WS-X6324-100FX-M switch	1q4t	2q2t
WS-X6224-100FX-MT switch	1q4t	2q2t
WS-X6548-RJ-21	1p1q0t	1p3q1t
WS-X6548-RJ-45	1p1q0t	1p3q1t
WS-X6348-RJ-21	1q4t	2q2t
WS-X6348-RJ21V-Z switch	1q4t	2q2t
WS-X6348-RJ-45	1q4t	2q2t
WS-X6348-RJ-45V switch	1q4t	2q2t
WS-X6148-RJ-45V	1q4t	2q2t
WS-X6148-RJ21V-D switch	1q4t	2q2t
WS-X6248-RJ-45	1q4t	2q2t
WS-X6248A-TEL	1q4t	2q2t
WS-X6248-TEL	1q4t	2q2t
WS-X6024-10FL-MT	1q4t	2q2t

Configuratie-, monitor- en voorbeelduitvoerplanning op Catalyst 6500/6000

Configuratie

In dit gedeelte worden alle stappen beschreven die nodig zijn om uitvoerschema's te configureren op een Catalyst 6500/6000 die Cisco IOS-software draait. Zie case 1 voor de standaardinstellingen van Catalyst 6500/6000: QoS is ingeschakeld en er wordt een standaardparameter van dit document gebruikt.

De configuratie van Catalyst 6500/6000 impliceert de volgende vijf stappen:

1. QoS inschakelen
2. Stel elke mogelijke CoS-waarde (Class of Service) in een wachtrij en een drempelwaarde in (optioneel)

3. Het WRR-gewicht instellen (optioneel)
4. Het configureren van de buffers die aan elke rij worden toegewezen (optioneel)
5. Het drempelniveau voor elke wachtrij instellen (optioneel)

Opmerking: elk van deze stappen is optioneel, met uitzondering van Stap 1. U kunt besluiten de standaardwaarde voor een of meer parameters te laten.

Stap 1: QoS inschakelen

Schakel eerst QoS in. Onthoud dat QoS standaard uitgeschakeld is. Wanneer QoS wordt uitgeschakeld, heeft de CoS-mapping die u hebt ingesteld geen invloed op de uitkomst. Er is één rij die eerst op de FIFO-manier wordt geserveerd, en alle pakketten worden er laten vallen.

```
cosmos#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
cosmos(config)#mls qos

QoS is enabled globally
  Microflow policing is enabled globally

QoS global counters:
    Total packets: 552638
    IP shortcut packets: 0
    Packets dropped by policing: 0
    IP packets with TOS changed by policing: 0
    IP packets with COS changed by policing: 0
    Non-IP packets with CoS changed by policing: 0
```

Stap 2: Stel elke mogelijke CoS-waarde in een wachtrij en een drempel in

Voor alle wachtrijtypes, verdeel de CoS aan een rij en een drempel. De mapping die is gedefinieerd voor een 2q2t-type poort wordt niet toegepast op een 1p2q2t-poort. Ook wordt de mapping voor 2q2t toegepast op alle havens met een 2 q2t wachtrijmechanisme. Geef deze opdrachten voor **cos-map** onder de interface uit:

```
wrr-queue  cos-map Q_number_(1-2) threshold_number_(1-2) cos_value_1 cos_value_2
priority-queue cos-map Q_number_(always 1) cos_value_1 cos_value_2
```

Opmerking: elk van deze opdrachten moet op één regel staan.

U kunt de WRR-wachtrij afzonderlijk configureren. Als er een prioriteitswachtrij is, kunt u deze configureren met de opdracht **prioriteitswachtrij**.

Opmerking: de rijen zijn altijd genummerd beginnend met de rij met de laagste prioriteit mogelijk en eindigend met de rij met strikte prioriteit die beschikbaar is. Bijvoorbeeld:

- Wachtrij 1 is de WRR-wachtrij met lage prioriteit.
- Wachtrij 2 is de rij met hoge prioriteit WRR.
- Wachtrij 3 is de rij met hoge prioriteit.

Herhaal deze handeling voor alle soorten wachtrijen, of anders blijft de standaard CoS toewijzing behouden. Dit is een voorbeeldconfiguratie voor 1p2q2t:

```
cosmos#configure terminal
cosmos(config)#interface gigabitethernet 1/1
cosmos(config-if)#priority-queue cos-map 1 5
!--- Assign a CoS of 5 to priority queue. cos-map configured on: Gi1/1 Gi1/2 cosmos(config-
if)#wrr-queue cos-map 1 1 0 1
!--- Assign CoS 0 and 1 to the first threshold of low-priority WRR queue. cos-map configured on:
Gi1/1 Gi1/2 cosmos(config-if)#wrr-queue cos-map 1 2 2 3
!--- Assign CoS 2 and 3 to the second threshold of low-priority WRR queue. cos-map configured
on: Gi1/1 Gi1/2 cosmos(config-if)#wrr-queue cos-map 2 1 4 6
!--- Assign CoS 4 and 6 to the first threshold of high-priority WRR queue. cos-map configured
on: Gi1/1 Gi1/2 cosmos(config-if)#wrr-queue cos-map 2 2 7
!--- Assign CoS 7 to the first threshold of high-priority WRR queue. cos-map configured on:
Gi1/1 Gi1/2
```

Controleer de configuratie:

```
cosmos#show queueing interface gigabitethernet 1/1
!--- Output suppressed. queue thresh cos-map ------------------------------------- 1 1 0 1 1 2
2 3 2 1 4 6 2 2 7 3 1 5 !--- Output suppressed.
```

Stap 3: Het WRR-gewicht configureren

Configureer het WRR-gewicht voor de twee WRR-rijen. Geef deze interfaceopdracht op:

```
wrr-queue bandwidth weight_for_Q1 weight_for_Q2
```

Gewicht 1 heeft betrekking op rij 1, wat de WRR-wachtrij met lage prioriteit zou moeten zijn. Houd dit gewicht altijd lager dan gewicht 2. Het gewicht kan elke waarde tussen 1 en 255 bedragen. Gebruik deze formules om het percentage toe te wijzen:

- Op rij 1—[gewicht 1 / (gewicht 1 + gewicht 2)]
- Naar rij 2—[gewicht 2 / (gewicht 1 + gewicht 2)]

U moet het gewicht definiëren voor alle soorten wachtrijen. Deze gewichtstypen hoeven niet hetzelfde te zijn. Dit is een voorbeeld voor 2q2t, waar wachtrij 1 20% van de tijd en rij 2 80% van de tijd wordt bediend:

```
cosmos#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
cosmos(config)#interface gigabitethernet 1/1
cosmos(config-if)#wrr-queue bandwidth ?
  <1-255>  enter bandwidth weight between 1 and 255
cosmos(config-if)#wrr-queue bandwidth 20 80
!--- Queue 1 is served 20% of the time, and queue 2 is served !--- 80% of the time.
cosmos(config-if)#
```

Controleer de configuratie:

```
cosmos#show queueing interface gigabitethernet 1/1
Interface GigabitEthernet1/1 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
  Port is untrusted
  Default cos is 0
  Transmit queues [type = 1p2q2t]:
    Queue Id    Scheduling  Num of thresholds
```

```
    ----------------------------------------
       1         WRR low             2
       2         WRR high            2
       3         Priority            1

    WRR bandwidth ratios:   20[queue 1]  80[queue 2]
    queue-limit ratios:     90[queue 1]   5[queue 2]
!--- Output suppressed.
```

Opmerking: U kunt verschillende WRR-gewichten voor elke interface configureren als het niet mogelijk is om CatOS-software te gebruiken.

Stap 4: Configureer de buffers die aan elke wachtrij zijn toegewezen

U moet de verhouding voor verzendwachtrij definiëren. Dit bepaalt hoe de buffers onder de verschillende wachtrijen worden verdeeld.

```
wrr-queue queue-limit percentage_WRR_Q1 percentage_WRR_Q2
cosmos(config)#interface gigabitethernet 1/2
cosmos(config-if)#wrr-queue queue-limit 70 15
!--- Queue 1 has 70% of the buffers. !--- Queues 2 and 3 both have 15% of the buffers. queue-
limit configured on: Gi1/1 Gi1/2
```

Opmerking: Als de wachtrijcapaciteit van uw gigabit poort 1p1q2t is, moet u hetzelfde niveau gebruiken voor de prioriteitswachtrij en voor de WRR-wachtrij met hoge prioriteit. Deze niveaus kunnen om hardwareredenen niet verschillen. Alleen de bandbreedte voor de twee WRR-wachtrijen wordt ingesteld. U gebruikt automatisch de zelfde waarde voor de rij van de hoge prioriteit WRR en de rij van de strikte prioriteit, als er enig is.

Sommige wachtrijen types hebben geen stembare rijgrootte. Een voorbeeld is 1p3q1t, dat beschikbaar is op WS-X6548RJ45. Deze wachtrijtypes zijn vast, en u kunt ze niet aanpassen.

Controleer de configuratie:

```
cosmos#show queueing interface gigabitethernet 1/2
Interface GigabitEthernet1/2 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
  Port is untrusted
  Default cos is 0
  Transmit queues [type = 1p2q2t]:
    Queue Id    Scheduling  Num of thresholds
    ----------------------------------------
       1         WRR low             2
       2         WRR high            2
       3         Priority            1

    WRR bandwidth ratios:    5[queue 1] 255[queue 2]
    queue-limit ratios:     70[queue 1]  15[queue 2]
```

Opmerking: Het is het beste om het grootste deel van de buffers achter te laten voor de WRR-wachtrij met lage prioriteit. Dit is de rij waar u extra buffers nodig hebt. De andere rijen worden bediend met een hogere prioriteit.

Stap 5: Het drempelniveau voor elke wachtrij configureren

Als laatste stap moet u het drempelniveau voor de WRED-wachtrij of de wachtrij voor staart configureren. Deze lijst bevat de opdrachten:

- Voor wachtrijen die WRED als valmechanisme voor de drempel gebruiken, geeft u deze opdrachten uit:

  ```
  wrr-queue random-dtect min-threshold Q_number threshold_1_value threshold_2_value
  wrr-queue random-dtect max-threshold Q_number threshold_1_value threshold_2_value
  ```

  Opmerking: elk van deze opdrachten moet op één regel staan.
- Voor wachtrijen die munt als valmechanisme gebruiken, geeft u deze opdracht uit:

  ```
  wrr-queue threshold Q_number threshold_1_value threshold_2_value
  ```

  Opmerking: deze opdracht moet op één regel staan.

Configuratie voor een WRED-wachtrij:

```
cosmos(config)#interface gigabitethernet 1/1
cosmos(config-if)#wrr-queue random-detect min-threshold  1 20 50
!--- This sets the threshold of queue 1 to 20 and 50% minimum threshold !--- configured on Gi1/1
Gi1/2. cosmos(config-if)#wrr-queue random-detect min-threshold  2 20 50
!--- This sets the threshold of queue 2 to 20 and 50% minimum threshold !--- configured on Gi1/1
Gi1/2. cosmos(config-if)#wrr-queue random-detect max-threshold  1 50 80
!--- This sets the threshold of queue 1 to 50 and 80% maximum threshold !--- configured on Gi1/1
Gi1/2. cosmos(config-if)#wrr-queue random-detect max-threshold  2 40 60
!--- This sets the threshold of queue 2 to 49 and 60% maximum threshold !--- configured on Gi1/1
Gi1/2.
```

Configuratie voor een uitrolrij:

```
cosmos(config)#interface fastethernet 3/1
cosmos(config-if)#wrr-queue threshold ?
  <1-2> enter threshold queue id (1-2)
cosmos(config-if)#wrr-queue threshold 1 ?
  <1-100>  enter percent of queue size between 1 and 100
cosmos(config-if)#wrr-queue threshold 1 50 100
!--- This sets the tail drop threshold for this 2q2t interface for !--- queue 1 (low-priority)
to 50 and 100% of the buffer. threshold configured on: Fa3/1 Fa3/2 Fa3/3 Fa3/4 Fa3/5 Fa3/6 Fa3/7
Fa3/8 Fa3/9 Fa3/10 Fa3/11 Fa3/12 cosmos(config-if)# cosmos(config-if)# cosmos(config-if)#wrr-
queue threshold 2 40 100
!--- This sets the tail drop threshold for this 2q2t interface for !--- queue 2 (high-priority)
to 40 and 100% of the buffer. threshold configured on: Fa3/1 Fa3/2 Fa3/3 Fa3/4 Fa3/5 Fa3/6 Fa3/7
Fa3/8 Fa3/9 Fa3/10 Fa3/11 Fa3/12 cosmos(config-if)#
```

Controleer de configuratie:

```
cosmos#show queueing interface gigabitethernet 1/1
Interface GigabitEthernet1/1 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
  Port is untrusted
  Default cos is 0
  Transmit queues [type = 1p2q2t]:
    Queue Id    Scheduling  Num of thresholds
    -----------------------------------------
       1         WRR low            2
       2         WRR high           2
       3         Priority           1
```

```
    WRR bandwidth ratios:   20[queue 1]  80[queue 2]
    queue-limit ratios:     70[queue 1]  15[queue 2]


    queue random-detect-min-thresholds
    ----------------------------
      1    20[1] 50[2]
      2    20[1] 50[2]

    queue random-detect-max-thresholds
    ---------------------------------
      1    50[1] 80[2]
      2    40[1] 60[2]

cosmos#show queueing interface fastethernet 3/1
Interface FastEthernet3/1 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
  Port is untrusted
  Default cos is 0
  Transmit queues [type = 2q2t]:
    Queue Id    Scheduling  Num of thresholds
    -----------------------------------------
       1         WRR low             2
       2         WRR high            2

    WRR bandwidth ratios:  100[queue 1] 255[queue 2]
    queue-limit ratios:     90[queue 1]  10[queue 2]

    queue tail-drop-thresholds
    --------------------------
    1     50[1] 100[2]
    2     40[1] 100[2]
```

U kunt de drempel niet configureren en de CoS niet per poort aan de wachtrij toewijzen. Alle wijzigingen worden toegepast op een reeks aaneengesloten havens:

- Vier poorten voor gigabit lijnkaarten—poorten 1 tot en met 4 zijn bij elkaar, en poorten 5 tot en met 8 zijn bij elkaar.
- Twaalf poorten voor 10/100 poorten of 100 glasvezel poorten op basis van 1q4t/2q2t wachtrij - 1 tot 12, 13 tot 24, 25 tot 36 en 36 tot 48.
- Om de exacte poort te bepalen die tot dezelfde ASIC behoort, gebruikt u de opdracht **Show interface mogelijkheden**.

Monitoruitvoer planning en configuratie controleren

Het makkelijkste bevel om uit te geven om de huidige run-tijd configuratie voor een poort met betrekking tot uitvoerschema te verifiëren is de **show Wachtende interface {gigabitethernet | fastethernet}** ***sleuf/poort*** opdracht. Deze opdracht geeft het type wachtrijen in de poort weer, de toewijzing van CoS aan de verschillende wachtrijen en drempels, het delen van de buffer en het WRR-gewicht. Hier is het 20% WRR voor rij 1 en 80% WRR voor rij 2. De opdracht geeft ook alle geconfigureerde informatie voor uitvoerschema weer en het aantal pakketten dat in elke wachtrij voor elke drempelwaarde wordt gedropt:

```
cosmos#show queueing interface gigabitethernet 1/1
Interface GigabitEthernet1/1 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
```

```
Port is untrusted
Default COS is 0
Transmit queues [type = 1p2q2t]:
  Queue Id    Scheduling  Num of thresholds
  ---------------------------------------
     1         WRR low            2
     2         WRR high           2
     3         Priority           1
  WRR bandwidth ratios:   20[queue 1]  80[queue 2]
  queue-limit ratios:     70[queue 1]  15[queue 2]

  queue random-detect-max-thresholds
  ----------------------------------
    1    50[1] 80[2]
    2    40[1] 60[2]

  queue thresh cos-map
  ---------------------------------------
  1     1      0 1
  1     2      2 3
  2     1      4 6
  2     2      7
  3     1      5

Receive queues [type = 1p1q4t]:
  Queue Id    Scheduling  Num of thresholds
  ---------------------------------------
     1         Standard           4
     2         Priority           1


  queue tail-drop-thresholds
  --------------------------
  1     100[1] 100[2] 100[3] 100[4]

  queue thresh cos-map
  ---------------------------------------
  1     1      0 1
  1     2      2 3
  1     3      4
  1     4      6 7
  2     1      5


Packets dropped on Transmit:
  BPDU packets:  0

  queue thresh    dropped  [cos-map]
  ---------------------------------------------------
  1     1              0  [0 1 ]
  1     2              0  [2 3 ]
  2     1              0  [4 6 ]
  2     2              0  [7 ]
  3     1              0  [5 ]

Packets dropped on Receive:
  BPDU packets:  0

  queue thresh    dropped  [cos-map]
  ---------------------------------------------------
  1     1              0  [0 1 ]
  1     2              0  [2 3 ]
  1     3              0  [4 ]
  1     4              0  [6 7 ]
```

```
    2     1              0  [5 ]
```

Voorbeeld van uitvoerplanning

Dit verkeer wordt ingespoten op Catalyst 6500/6000:

- In poort gigabit 1/2: één gigabit verkeer met voorrang van nul
- In poort gigabit 5/2:133 MB verkeer met voorrang van zeven133 MB verkeer met voorrang van zes133 MB verkeer met voorrang van vijf133 MB verkeer met voorrang van vier133 MB verkeer met voorrang van drie133 MB verkeer met voorrang van twee133 MB verkeer met voorrang van één

Al het eenastverkeer verlaat de switch per poort gigabit 1/1, wat erg overtekend is.

Zaak 1: QoS is ingeschakeld en er wordt een standaard parameter gebruikt

Het opdracht van de **show een wachtrij voor een interface met Gigabit Ethernet 1/1** vormt alle uitvoer in dit voorbeeld. Deze opdracht bevat aanvullende informatie over invoerschema's. Aangezien dit document echter alleen uitvoerschema's bevat, wordt deze uitvoer onderdrukt.

Wanneer QoS mondiaal is ingeschakeld en alle standaardparameters in gebruik zijn, levert dit resultaat na een paar minuten op:

```
nelix#show queueing interface gigabitethernet 1/1
Interface GigabitEthernet1/1 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
  Trust state: trust DSCP
  Default cos is 0
  Transmit queues [type = 1p2q2t]:
    Queue Id    Scheduling  Num of thresholds
    -----------------------------------------
       1         WRR low             2
       2         WRR high            2
       3         Priority            1

    WRR bandwidth ratios:  100[queue 1] 255[queue 2]
    queue-limit ratios:     90[queue 1]   5[queue 2]

    queue random-detect-max-thresholds
    ----------------------------------
      1    40[1] 100[2]
      2    40[1] 100[2]

    queue thresh cos-map
    ---------------------------------------
    1     1      0 1
    1     2      2 3
    2     1      4
    2     2      6 7
    3     1      5

  Packets dropped on Transmit:
    BPDU packets:  0

    queue thresh    dropped  [cos-map]
    ---------------------------------------------------
    1     1       149606424  [0 1 ]
    1     2               0  [2 3 ]
```

```
    2     1        16551394  [4 ]
    2     2         4254446  [6 7 ]
    3     1               0  [5 ]
```

In deze uitvoer zijn de standaardwaarden:

- WRR-gewicht voor rij 1-100 / (100 + 255) = 28%
- WRR-gewicht voor rij 2-255 / (255 + 100) = 72%
- Buffer delen: —90% voor rij 1, 5% voor rij 2 en 5% voor rij 2 met strikte prioriteit

De meeste pakketten in de rij met lage prioriteit WRR worden gedropt, maar sommige worden nog steeds in de rij met hoge prioriteit WRR voor beide drempels gedropt. In totaal zijn er 170.412.264 druppels (149.606.424 + 16.551.394 + 4.254.446). Deze druppels zijn als volgt verdeeld:

- 149.606.424 / 170.412.264 = 88% van de druppels in rij 1 (eerste drempelpakket met CoS 0 en 1)
- 16.551.394 / 170.412.264 = 10% van de druppels in rij 2 (eerste drempelpakket met CoS 4)
- 4.254.446 / 170.412.264 = 2% van de druppels in rij 2 (tweede drempelpakket met CoS van 6 of 7)

Opmerking: je ziet geen druppels in de rij met strikte prioriteit.

Zaak 2: WRR-gewicht wijzigen

Zoals opgemerkt in zaak 1: QoS is ingeschakeld en een Standaardparameter wordt gebruikt, worden pakketten in rij 2 nog steeds verwijderd. Wijzig het WRR gewicht om meer bandbreedte te geven aan rij 2. Nu wordt rij 1 geleegd 4 procent van de tijd en rij 2 geleegd 96 procent van de tijd:

```
show run interface gigabitethernet 1/1
interface GigabitEthernet1/1
 no ip address
 wrr-queue bandwidth 10 255
 mls qos trust dscp
 switchport
 switchport mode access
end

nelix#show queueing interface gigabitethernet 1/1
Interface GigabitEthernet1/1 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
  Trust state: trust DSCP
  Default cos is 0
  Transmit queues [type = 1p2q2t]:
    Queue Id    Scheduling  Num of thresholds
    -----------------------------------------
       1         WRR low             2
       2         WRR high            2
       3         Priority            1

    WRR bandwidth ratios:   10[queue 1] 255[queue 2]
    queue-limit ratios:     90[queue 1]   5[queue 2]

    queue random-detect-max-thresholds
    ----------------------------------
      1    40[1] 100[2]
      2    40[1] 100[2]

    queue thresh cos-map
```

```
    ---------------------------------------
    1     1      0 1
    1     2      2 3
    2     1      4
    2     2      6 7
    3     1      5

  Packets dropped on Transmit:
    BPDU packets:  0

    queue thresh    dropped  [cos-map]
    ---------------------------------------------------
    1     1         2786205  [0 1 ]
    1     2               0  [2 3 ]
    2     1           11363  [4 ]
    2     2              69  [6 7 ]
    3     1               0  [5 ]
```

Zoals in deze output wordt gezien, is het percentage druppels in rij 2 nu veel lager. In totaal worden 2.797.637 druppels op deze manier verdeeld:

- 2.786.205 / 2.797.637 = 99.591% van de druppels in rij 1 (met pakket CoS 0 en 1)
- 11.363 / 2.797.637 = 0,408% van de druppels in rij 2 (eerste drempel met pakketcode CoS 4)
- 69 / 2.797.637 = 0,001% van de druppels in rij 2 (tweede drempel voor verpakkingen met CoS 6 en 7)

Als u verschillende WRR-gewichten gebruikt, zorgt dit voor meer QoS in rij 2.

Zaak 3: Aanvullende WRR-gewichtswijziging

Je kunt nog agressiever zijn met het WRR-gewicht. In deze steekproefuitvoer wordt slechts 0,39% van het gewicht aan rij 1 gegeven:

```
show run interface gigabitethernet 1/1
interface GigabitEthernet1/1
 no ip address
 wrr-queue bandwidth 1 255
 mls qos trust dscp
 switchport
 switchport mode access
end

nelix#show queueing interface gigabitethernet 1/1
Interface GigabitEthernet1/1 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
  Trust state: trust DSCP
  Default cos is 0
  Transmit queues [type = 1p2q2t]:
    Queue Id    Scheduling  Num of thresholds
    ---------------------------------------
       1         WRR low             2
       2         WRR high            2
       3         Priority            1

    WRR bandwidth ratios:    1[queue 1] 255[queue 2]
    queue-limit ratios:     90[queue 1]   5[queue 2]

    queue random-detect-max-thresholds
    ----------------------------------
```

```
      1    40[1] 100[2]
      2    40[1] 100[2]

    queue thresh cos-map
    ---------------------------------------
    1     1      0 1
    1     2      2 3
    2     1      4
    2     2      6 7
    3     1      5

  Packets dropped on Transmit:
    BPDU packets:  0

    queue thresh    dropped  [cos-map]
    ---------------------------------------------------
    1     1         2535315  [0 1 ]
    1     2               0  [2 3 ]
    2     1             705  [4 ]
    2     2              73  [6 7 ]
    3     1               0  [5 ]
```

Zelfs met het agressieve WRR gewicht, worden er nog steeds pakketten in rij 2 geworpen. In vergelijking is het echter niet veel pakketten. Er is nu slechts een 0,03 procent pakketdaling in rij 2.

Zaak 4: Wachtrij limiteren toewijzing

Zoals blijkt uit zaak 2: WRR-gewicht wijzigen en case 3: Aanvullende WRR Gebiedswijzigingen, pakketten vallen nog steeds in rij 2, hoewel het WRR-percentage u verzekert dat de daling minimaal is. Wanneer echter de tweede drempel (die op 100 procent wordt ingesteld) in rij 2 wordt bereikt, worden sommige pakketten nog steeds niet geworpen.

Om dit te verbeteren, wijzigt u de wachtrijlimiet (grootte van de buffer die aan elke wachtrij is toegewezen). In dit voorbeeld is de wachtrijlimiet ingesteld op 70 procent voor rij 1, 15 procent voor rij 2 en 15 procent voor rij 2.

```
show run gigabitethernet 1/1
interface GigabitEthernet1/1
 no ip address
 wrr-queue bandwidth 1 255
 wrr-queue queue-limit 70 15
 mls qos trust dscp
 switchport
 switchport mode access
end

nelix#show queueing interface gigabitethernet 1/1
Interface GigabitEthernet1/1 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
  Trust state: trust DSCP
  Default cos is 0
  Transmit queues [type = 1p2q2t]:
    Queue Id    Scheduling  Num of thresholds
    -----------------------------------------
       1         WRR low             2
       2         WRR high            2
       3         Priority            1

    WRR bandwidth ratios:    1[queue 1] 255[queue 2]
```

```
  queue-limit ratios:     70[queue 1]  15[queue 2]

  queue random-detect-max-thresholds
  ----------------------------------
    1    40[1] 100[2]
    2    40[1] 100[2]

  queue thresh cos-map
  --------------------------------------
  1     1      0 1
  1     2      2 3
  2     1      4
  2     2      6 7
  3     1      5

Receive queues [type = 1p1q4t]:
  Queue Id    Scheduling  Num of thresholds
  -----------------------------------------
     1         Standard            4
     2         Priority            1

  queue tail-drop-thresholds
  --------------------------
  1     100[1] 100[2] 100[3] 100[4]

  queue thresh cos-map
  --------------------------------------
  1     1      0 1
  1     2      2 3
  1     3      4
  1     4      6 7
  2     1      5

Packets dropped on Transmit:
  BPDU packets:  0

  queue thresh    dropped  [cos-map]
  ---------------------------------------------------
  1     1       154253046  [0 1 ]
  1     2               0  [2 3 ]
  2     1               0  [4 ]
  2     2               0  [6 7 ]
  3     1               0  [5 ]
```

De druppels komen alleen voor in rij 1.

Gebruik uitvoerplanning om vertraging en jitter te verminderen

De casestudy's in het gedeelte Voorbeeld van uitvoerplanning tonen het voordeel aan van het uitvoeren van uitvoerschema's om een daling van VoIP of missie-kritiek verkeer te vermijden in het geval van overabonnement op de uitvoerpoort. Overabonnement komt niet zeer vaak voor in een normaal netwerk (vooral op een gigabit link). Overabonnement dient alleen plaats te vinden tijdens piektijden of tijdens doorbraken die zich binnen een zeer korte periode voordoen.

Zelfs zonder enige overabonnement kan uitvoerschema's van grote hulp zijn in een netwerk waar QoS van eind tot eind wordt geïmplementeerd. Dit gedeelte geeft voorbeelden van de manier waarop uitvoerschema's vertraging en jitter kunnen helpen verminderen.

Vertraging verminderen

De vertraging van een pakje neemt toe vanwege de tijd die "verloren" is in de buffer van elke switch terwijl het wachten om te worden verzonden. Een klein spraakpakket met een CoS van 5 wordt bijvoorbeeld tijdens een grote back-up of bestandsoverdracht via een poort verzonden. Stel dat er geen QoS is voor de uitvoerpoort en dat het kleine spraakpakket na 10 grote pakketten met 1500 bytes in de wachtrij wordt geplaatst. In dit geval, kunt u gemakkelijk berekenen dat de gigabit snelheid tijd die nodig is om de 10 grote pakketten te verzenden is:

- (10 x 1500 x 8) = 120.000 bits doorgegeven in 120 microseconden

Als dit pakket acht of negen switches moet overschrijden terwijl u door het netwerk passeert, kan een vertraging van ongeveer 1 milliseconde resulteren. Dit omvat slechts vertragingen in de uitvoerwachtrij van de switch die in het netwerk wordt doorgehaald.

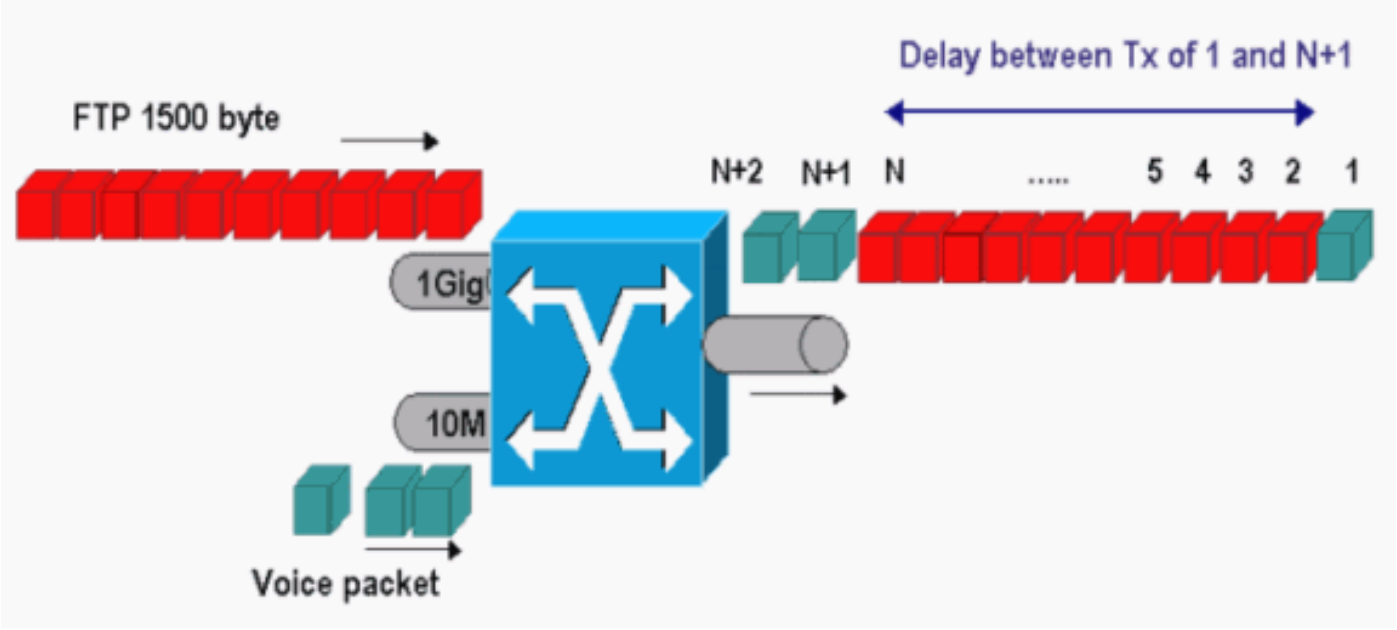
Opmerking: Als u dezelfde 10 grote pakketten op een 10MB-interface in een rij moet zetten (bijvoorbeeld verbonden met een IP-telefoon en een PC), is de startvertraging:

- (10 x 1500 x 8) = 120.000 bits verzonden in 12 milliseconden

De implementatie van uitvoerschema waarborgt dat spraakpakketten met een CoS van 5 in de wachtrij met strikte prioriteit worden geplaatst en vóór elke pakketten met een CoS van minder dan 5 worden verzonden. Dit vermindert vertraging.

Jitter verminderen

Een ander belangrijk voordeel van het plannen van de productie is het verminderen van de jitter. Jitter is de variatie in vertraging voor pakketten binnen dezelfde stroom. Dit voorbeeldscenario toont hoe de uitvoerplanning jitter kan verminderen:



In dit scenario moet dezelfde uitvoerpoort twee stromen verzenden:

- Eén inkomende spraakstroom op een 10 MB Ethernet-poort.
- Eén inkomende FTP-stream op een 1 Gigabit Ethernet uplinks-poort.

Beide stromen verlaten de switch door dezelfde uitvoerpoort. Dit voorbeeld toont wat zonder het gebruik van uitvoerschema kan voorkomen. Alle grote gegevenspakketten kunnen tussen twee spraakpakketten worden uitgewisseld. Dit creëert jitter in de ontvangst van het stempakket van de zelfde stroom. Er is een grotere vertraging tussen de ontvangst van pakket 1 en *pakket* n + 1

wanneer de switch het grote gegevenspakket doorgeeft. De vertraging tussen *n* + 1 en *n* + 2 is echter verwaarloosbaar. Dit resulteert in jitter in de stroom van het spraakverkeer. U kunt dit probleem met het gebruik van een rij met strikte prioriteit gemakkelijk vermijden. Zorg ervoor dat u de CoS waarde van de spraakpakketten in de wachtrij met strikte prioriteit instelt.

Conclusie

In dit document hebt u casestudy's gezien van hoe u uitvoerwachtrij voor een Catalyst 6500/6000 kunt configureren en oplossen die Cisco IOS-software draait. U hebt ook de voordelen van output planning in de meeste netwerken met stemverkeer gezien:

- Vermijd de daling van kritisch verkeer in het geval van overinschrijving op de uitvoerpoort.
- Vermindert de vertraging.
- Vermindert de jitter.

Gerelateerde informatie

- QoS O-planning bij Catalyst 6500/6000 Series Switches die CatOS-systeemsoftware uitvoeren
- Inzicht op Quality-of-Service op Catalyst 6000 Series Switches
- Productondersteuningspagina's voor LAN
- Ondersteuningspagina voor LAN-switching
- Technische ondersteuning en documentatie – Cisco Systems