

# Programação de saída de QoS em Switches Catalyst 6500/6000 Series executando o Cisco IOS System Software

## Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Conventions](#)

[Informações de Apoio](#)

[Quedas da fila da saída](#)

[Recurso de enfileiramento de saída de diferentes placas de linha no Catalyst 6500/6000](#)

[Entender o recurso de fila de uma porta](#)

[Configuração, monitor e exemplo de programação de saída no Catalyst 6500/6000](#)

[Configuração](#)

[Monitorar o Agendamento de Saída e Verificar Configurações](#)

[Exemplo de programação do emissor](#)

[Usar o Agendamento de Saída para Reduzir o Atraso e o Jitter](#)

[Reduza o atraso](#)

[Reduzir o atraso de sincronismo](#)

[Conclusão](#)

[Informações Relacionadas](#)

## Introduction

O uso da programação de saída garante que o tráfego importante não caia no caso de um grande excesso de assinaturas. Este documento discute todas as técnicas e algoritmos envolvidos na programação de saída no switch Catalyst 6500/6000. Este documento também explica como configurar e verificar a operação da programação de saída no Catalyst 6500/6000 que executa o Cisco IOS® Software.

Consulte [Programação de Saída de QoS em Catalyst 6500/6000 Series Switches executando CatOS System Software](#) para obter mais informações sobre Detecção Antecipada Aleatória Ponderada (WRED - Weighted Random Early Detection), Rodízio Ponderado (WRR - Weighted Round Robin) e queda traseira.

## Prerequisites

## Requirements

Não existem requisitos específicos para este documento.

## Componentes Utilizados

Este documento não se restringe a versões de software e hardware específicas.

## Conventions

Consulte as [Convenções de Dicas Técnicas da Cisco para obter mais informações sobre convenções de documentos](#).

## Informações de Apoio

### Quedas da fila da saída

As quedas de saída são causadas por uma interface congestionada. Uma causa comum disso pode ser o tráfego de um link de largura de banda alta que é comutado para um link de largura de banda mais baixa ou o tráfego de vários links de entrada que é comutado para um único link de saída.

Por exemplo, se uma quantidade grande de tráfego intermitente chegar em uma interface gigabit e for comutada para uma interface de 100 Mbps, isso poderá causar cancelamentos de saída para aumentar a interface de 100 Mbps. Isso ocorre porque a fila de saída nessa interface está sobrecarregada pelo excesso de tráfego devido à incompatibilidade de velocidade entre as larguras de banda de entrada e saída. A taxa de tráfego na interface de saída não pode aceitar todos os pacotes que devem ser enviados.

Para resolver o problema, a melhor solução é aumentar a velocidade da linha. No entanto, há maneiras de evitar, diminuir ou controlar quedas de saída quando você não deseja aumentar a velocidade da linha. Você pode evitar quedas de saída somente se quedas de saída forem uma consequência de rajadas curtas de dados. Se as quedas de saída forem causadas por um fluxo constante de alta taxa, você não poderá impedir as quedas. Entretanto, você pode controlá-los.

## Recurso de enfileiramento de saída de diferentes placas de linha no Catalyst 6500/6000

Se você não tiver certeza sobre o recurso de enfileiramento de uma porta, emita o comando **show queueing interface {gigabitethernet comando *mod/port* / fastethernet**. Aqui são mostradas as primeiras linhas de saída de um comando **show queueing interface**. A porta está em uma placa de linha do Supervisor Engine 1A:

```
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.

A saída mostra que essa porta tem um tipo de enfileiramento de saída conhecido como 1p2q2t.

Outra maneira de ver o tipo de enfileiramento disponível em uma porta específica é emitir o comando **show interface capabilities**:

```

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

```

## Entender o recurso de fila de uma porta

Há vários tipos de filas disponíveis nos switches Catalyst 6500/6000. Esta tabela explica a notação da arquitetura de QoS da porta:

Transmissão (Tx)/Recepção (Rx)	Notação da fila	Número de filas	priority que ue	Nº de filas WRR	Nº e tipo de limite para filas WRR
Tx	2q2t	2	—	2	2 queda traseira configurável
Tx	1p2q2t	3	1	2	2 WRED configurável
Tx	1p3q1t	4	1	3	1 WRED configurável
Tx	1p2q1t	3	1	2	1 WRED configurável
Rx	1q4t	1	—	—	4 queda traseira configurável
Rx	1p1q4t	2	1	1	4 queda traseira configurável
Rx	1p1q	2	1	1	Não configurável

	0t				
Rx	1p1q 8t	2	1	1	8 WRED configurável
Tx	1p3q 8t	4	1	3	8 WRED configurável ou queda traseira
Tx	1p7q 8t	8	1	7	8 WRED configurável ou queda traseira
Rx	1q2t	1	—	—	1 queda traseira configurável = 1 não configurável
Rx	1q8t	1	—	—	8 queda traseira configurável
Rx	2q8t	2	—	2	8 queda traseira configurável

A tabela a seguir lista alguns dos módulos e tipos de fila nos lados Rx e Tx da interface ou porta. Se o seu módulo não estiver listado aqui, use o comando **show interface capabilities** para determinar a capacidade da fila disponível. O comando **show interface capabilities** é descrito na [Output Queueing Capability of Different Line Cards](#) na seção [Catalyst 6500/6000](#).

Módulo	Filas Rx	Filas Tx
WS-X6K-S2-PFC2	1p1q4t	1p2q2t
WS-X6K-SUP1A-2GE	1p1q4t	1p2q2t
WS-X6K-SUP1-2GE	1q4t	2q2t
WS-X6501-10GEX4	1p1q8t	1p2q1t
WS-X6502-10GE	1p1q8t	1p2q1t
WS-X6516-GBIC	1p1q4t	1p2q2t
WS-X6516-GE-TX	1p1q4t	1p2q2t
WS-X6416-GBIC	1p1q4t	1p2q2t
WS-X6416-GE-MT	1p1q4t	1p2q2t
WS-X6316-GE-TX	1p1q4t	1p2q2t
WS-X6408A-GBIC	1p1q4t	1p2q2t
WS-X6408-GBIC	1q4t	2q2t
WS-X6524-100FX-MM	1p1q0t	1p3q1t
WS-X6324-100FX-SM	1q4t	2q2t
WS-X6324-100FX-MM	1q4t	2q2t
WS-X6224-100FX-MT	1q4t	2q2t
WS-X6548-RJ-21	1p1q0t	1p3q1t
WS-X6548-RJ-45	1p1q0t	1p3q1t
WS-X6348-RJ-21	1q4t	2q2t
WS-X6348-RJ21V	1q4t	2q2t
WS-X6348-RJ-45	1q4t	2q2t
WS-X6348-RJ-45V	1q4t	2q2t

WS-X6148-RJ-45V	1q4t	2q2t
WS-X6148-RJ21V	1q4t	2q2t
WS-X6248-RJ-45	1q4t	2q2t
WS-X6248A-TEL	1q4t	2q2t
WS-X6248-TEL	1q4t	2q2t
WS-X6024-10FL-MT	1q4t	2q2t

## [Configuração, monitor e exemplo de programação de saída no Catalyst 6500/6000](#)

### [Configuração](#)

Esta seção descreve todas as etapas necessárias para configurar o agendamento de saída em um Catalyst 6500/6000 que executa o Cisco IOS Software. Para a configuração padrão do Catalyst 6500/6000, consulte o [Caso 1: A QoS está habilitada e um parâmetro padrão é usado](#) na seção deste documento.

A configuração do Catalyst 6500/6000 implica estas cinco etapas:

1. [Habilitar QoS](#)
2. [Mapear cada valor de classe de serviço \(CoS\) possível para uma fila e um limite](#) (opcional)
3. [Configurar o peso do WRR](#) (opcional)
4. [Configurar os buffers atribuídos a cada fila](#) (opcional)
5. [Configurar o nível de limite para cada fila](#) (opcional)

**Observação:** cada uma dessas etapas é opcional, com exceção da Etapa 1. Você pode decidir deixar o valor padrão para um ou mais parâmetros.

### [Passo 1: Habilitar QoS](#)

Primeiro, habilite a QoS. Lembre-se de que a QoS está desativada, por padrão. Quando a QoS é desabilitada, o mapeamento de CoS configurado não afeta o resultado. Há uma fila atendida de maneira FIFO (first in, first out, primeiro a entrar, primeiro a sair) e todos os pacotes são descartados lá.

```
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
```

### [Passo 2: Mapear cada valor de CoS possível para uma fila e um limite](#)

Para todos os tipos de fila, atribua o CoS a uma fila e a um limite. O mapeamento definido para um tipo de porta 2q2t não é aplicado a nenhuma porta 1p2q2t. Além disso, o mapeamento para 2q2t é aplicado a todas as portas com um mecanismo de enfileiramento 2q2t. Emita estes comandos **cos-map** na interface:

```
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
```

**Observação:** cada um desses comandos deve estar em uma linha.

Você pode configurar separadamente a fila WRR. Se houver uma fila de prioridade, você poderá configurá-la com o comando **priority-queue**.

**Observação:** as filas são sempre numeradas começando com a fila de prioridade mais baixa possível e terminando com a fila de prioridade estrita disponível. Por exemplo:

- A fila 1 é a fila WRR de baixa prioridade.
- A fila 2 é a fila WRR de alta prioridade.
- A fila 3 é a fila de prioridade estrita.

Repita essa operação para todos os tipos de filas ou a atribuição de CoS padrão permanece. Este é um exemplo de configuração para 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
```

**Verifique a configuração:**

```
cosmos#show queuing 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.
```

### [Passo 3: Configurar o peso do WRR](#)

Configure o peso WRR para as duas filas WRR. Emita este comando de interface:

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

O peso 1 se refere à fila 1, que deve ser a fila WRR de baixa prioridade. Mantenha sempre este peso um nível inferior ao peso 2. O peso pode ter qualquer valor entre 1 e 255. Use estas fórmulas para atribuir a porcentagem:

- Para a fila 1—[peso 1 / (peso 1 + peso 2)]
- Para a fila 2—[peso 2 / (peso 1 + peso 2)]

Você deve definir o peso para todos os tipos de filas. Esses tipos de peso não precisam ser os mesmos. Este é um exemplo para 2q2t, onde a fila 1 é atendida 20% do tempo e a fila 2 é atendida 80% do tempo:

```
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)#
```

**Verifique a configuração:**

```
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 = lp2q2t]:
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.
```

**Observação:** você pode configurar diferentes pesos WRR para cada interface quando não for possível usar o CatOS Software.

#### Passo 4: Configure os buffers atribuídos a cada fila

Você deve definir a proporção da fila de transmissão. Isso determina como os buffers são divididos entre as diferentes filas.

```
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
```

**Observação:** se o recurso de enfileiramento de sua porta gigabit for 1p1q2t, você precisará usar o mesmo nível para a fila de prioridade estrita e para a fila WRR de alta prioridade. Esses níveis não podem ser diferentes por razões de hardware. Somente a largura de banda das duas filas WRR está configurada. Você usa automaticamente o mesmo valor para a fila WRR de alta prioridade e a fila de prioridade estrita, se houver.

Alguns tipos de enfileiramento não têm tamanho de fila ajustável. Um exemplo é o 1p3q1t, que está disponível no WS-X6548RJ45. Esses tipos de fila são fixos e você não pode modificá-los.

## Verifique a configuração:

```
cosmos#show queuing interface gigabitethernet 1/2
Interface GigabitEthernet1/2 queuing strategy:  Weighted Round-Robin
Port QoS is enabled
Port is untrusted
Default cos is 0
Transmit queues [type = lp2q2t]:
  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]
```

**Observação:** é melhor deixar a maior parte dos buffers para a fila WRR de baixa prioridade. Esta é a fila onde você precisa ativar o buffer adicional. As outras filas são servidas com prioridade mais elevada.

## [Passo 5: Configurar o nível de limite para cada fila](#)

Como etapa final, configure o nível de limite para a fila WRED ou para a fila de queda traseira. Esta lista fornece os comandos:

- Para as filas que usam WRED como mecanismo de queda para o limiar, emita estes comandos:

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

**Observação:** cada um desses comandos deve estar em uma linha.

- Para as filas que usam queda traseira como mecanismo de queda, emita este comando:

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

**Observação:** esse comando deve estar em uma linha.

## Configuração de uma fila WRED:

```
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.
```

## Configuração para uma fila de queda traseira:

```
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)#

```

## Verifique a configuração:

```

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 = lp2q2t]:
  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]

```

Você não pode configurar o limite e atribuir o CoS à fila por porta. Todas as alterações são aplicadas a um conjunto de portas contíguas:

- Quatro portas para placas de linha gigabit—as portas 1 a 4 estão juntas e as portas 5 a 8 estão juntas.
- Doze portas para 10/100 portas ou 100 portas de fibra baseadas no enfileiramento 1q4t/2q2t—1 a 12, 13 a 24, 25 a 36 e 36 a 48.
- Para determinar a porta exata que pertence ao mesmo ASIC, use o comando **show interface capabilities**.

## Monitorar o Agendamento de Saída e Verificar Configurações

O comando mais fácil de executar para verificar a configuração atual de tempo de execução de uma porta em relação ao agendamento de saída é o comando **show queueing interface {gigabitethernet comando | fastethernet} slot/porta**. Esse comando exibe o tipo de enfileiramento na porta, o mapeamento de CoS para as diferentes filas e limiares, o compartilhamento de buffer e o peso de WRR. Aqui, é 20% de WRR para a fila 1 e 80% de WRR para a fila 2. O comando também exibe todas as informações configuradas para o agendamento de saída e o número de pacotes que são descartados em cada fila para cada limiar:

```
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 ]

```

## Exemplo de programação do emissor

Esse tráfego é injetado no Catalyst 6500/6000:

- Na porta gigabit 1/2: um gigabit de tráfego com precedência de zero
- Na porta gigabit 5/2: 133 MB de tráfego com precedência de sete  
133 MB de tráfego com precedência de seis  
133 MB de tráfego com precedência de cinco  
133 MB de tráfego com precedência de quatro  
133 MB de tráfego com precedência de três  
133 MB de tráfego com precedência de dois  
133 MB de tráfego com precedência de um

Todo o tráfego unicast sai do switch por porta gigabit 1/1, que está com excesso de assinaturas.

## Caso 1: A QoS está habilitada e um parâmetro padrão é usado

O comando **show queueing interface gigabitethernet 1/1** configura toda a saída neste exemplo. O comando fornece informações adicionais sobre o agendamento de entrada. No entanto, como este documento abrange somente o agendamento de saída, ele suprime essa saída.

Quando a QoS está habilitada globalmente e todos os parâmetros padrão estão em uso, essa saída resulta após alguns minutos:

```

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 = lp2q2t]:
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 ]
```

Nesta saída, os valores padrão são:

- Peso WRR para a fila 1— $100 / (100 + 255) = 28\%$
- Peso WRR para a fila 2— $255 / (255 + 100) = 72\%$
- Compartilhamento de buffer: —90% para a fila 1, 5% para a fila 2 e 5% para a fila de prioridade estrita

A maioria dos pacotes na fila WRR de baixa prioridade são descartados, mas alguns ainda são descartados na fila WRR de alta prioridade para ambos os limiares. Há um total de 170.412.264 gotas ( $149.606.424 + 16.551.394 + 4.254.446$ ). Essas gotas são divididas da seguinte forma:

- $149.606.424 / 170.412.264 = 88\%$  das quedas na fila 1 (primeiro pacote de limiar com CoS 0 e 1)
- $16.551.394 / 170.412.264 = 10\%$  das quedas na fila 2 (primeiro pacote de limiar com CoS 4)
- $4.254.446 / 170.412.264 = 2\%$  de quedas na fila 2 (segundo pacote de limiar com CoS de 6 ou 7)

**Observação:** você não vê nenhum descarte na fila de prioridade estrita.

## Caso 2: Modificar Peso do WRR

Como observado no [caso 1: A QoS está habilitada e um parâmetro padrão é usado](#), pacotes na fila 2 ainda estão sendo descartados. Modifique o peso do WRR para dar mais largura de banda à fila 2. Agora, a fila 1 é esvaziada 4% do tempo, e a fila 2 é esvaziada 96% do tempo:

```
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 = lp2q2t]:

```

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

```

Queue Id	Threshold 1	Threshold 2
1	40[1]	100[2]
2	40[1]	100[2]

```

queue thresh cos-map

```

Queue Id	Threshold	CoS
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]

```

Queue Id	Threshold	Dropped	CoS
1	1	2786205	[0 1]
1	2	0	[2 3]
2	1	11363	[4]
2	2	69	[6 7]
3	1	0	[5]

Como visto nesta saída, o percentual de quedas na fila 2 agora é muito menor. Um total de 2.797.637 gotas é dividido desta forma:

- $2.786.205 / 2.797.637 = 99,591\%$  das quedas na fila 1 (com pacote de CoS 0 e 1)
- $11.363 / 2.797.637 = 0,408\%$  de quedas na fila 2 (primeiro limiar com pacote CoS 4)
- $69 / 2.797.637 = 0,001\%$  de quedas na fila 2 (segundo limiar para pacote com CoS 6 e 7)

Se você usa vários pesos WRR, ele garante mais QoS na fila 2.

### Caso 3: Modificação adicional de peso de WRR

Você pode ser ainda mais agressivo com o peso de WRR. Nesta saída de exemplo, apenas 0,39% do peso é dado à fila 1:

```
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 = lp2q2t]:
```

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]

Mesmo com o peso de WRR agressivo, os pacotes ainda estão sendo descartados na fila 2. No entanto, em comparação, não são muitos pacotes. Agora há apenas 0,03% de queda de pacote na fila 2.

#### [Caso 4: Modificar atribuição de buffer de limite de fila](#)

Como se pode ver no [caso 2: Modifique o peso do WRR](#) e o [caso 3: Seções adicionais de WRR Weight Modification](#), os pacotes ainda estão sendo descartados na fila 2, embora o percentual de WRR garanta que a queda é mínima. No entanto, quando o segundo limiar (que é definido como 100%) é atingido na fila 2, alguns pacotes continuam a ser descartados.

Para melhorar isso, altere o limite da fila (tamanho do buffer atribuído a cada fila). Neste exemplo,

o limite da fila é definido como 70% para a fila 1, 15% para a fila 2 e 15% para a fila de prioridade estrita:

**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 = lp2q2t]:

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 = lplq4t]:

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 ]

Agora, as quedas só ocorrem na fila 1.

## Usar o Agendamento de Saída para Reduzir o Atraso e o Jitter

Os estudos de casos na seção [Exemplo de Programação de Saída](#) demonstram o benefício da implementação do agendamento de saída para evitar uma queda de VoIP ou tráfego de missão crítica em caso de excesso de assinatura da porta de saída. A sobreassinatura não ocorre com muita frequência em uma rede normal (particularmente em um link gigabit). O excesso de assinaturas só deve ocorrer durante horários de pico de tráfego ou durante surtos de tráfego que ocorram em um período de tempo muito curto.

Mesmo sem excesso de assinaturas, o agendamento de saída pode ser de grande ajuda em uma rede onde a QoS é implementada de ponta a ponta. Esta seção fornece exemplos de como o agendamento de saída pode ajudar a reduzir o atraso e a instabilidade.

### Reduza o atraso

O atraso de um pacote aumenta devido ao tempo "perdido" no buffer de cada switch enquanto espera para ser transmitido. Por exemplo, um pequeno pacote de voz com um CoS de 5 é enviado de uma porta durante um grande backup ou transferência de arquivos. Suponha que não haja QoS para a porta de saída e que o pequeno pacote de voz seja enfileirado após 10 pacotes grandes de 1.500 bytes. Nesse caso, você pode facilmente calcular que o tempo de velocidade de gigabit necessário para transmitir os 10 pacotes grandes é:

- $(10 \times 1500 \times 8) = 120.000$  bits transmitidos em 120 microssegundos

Se esse pacote precisar cruzar oito ou nove switches ao passar pela rede, pode ocorrer um atraso de aproximadamente 1 milissegundo. Isso inclui apenas atrasos na fila de saída do switch cruzado na rede.

**Observação:** se você precisar colocar em fila os mesmos 10 pacotes grandes em uma interface de 10MB (por exemplo, conectados a um telefone IP e a um PC), o atraso será:

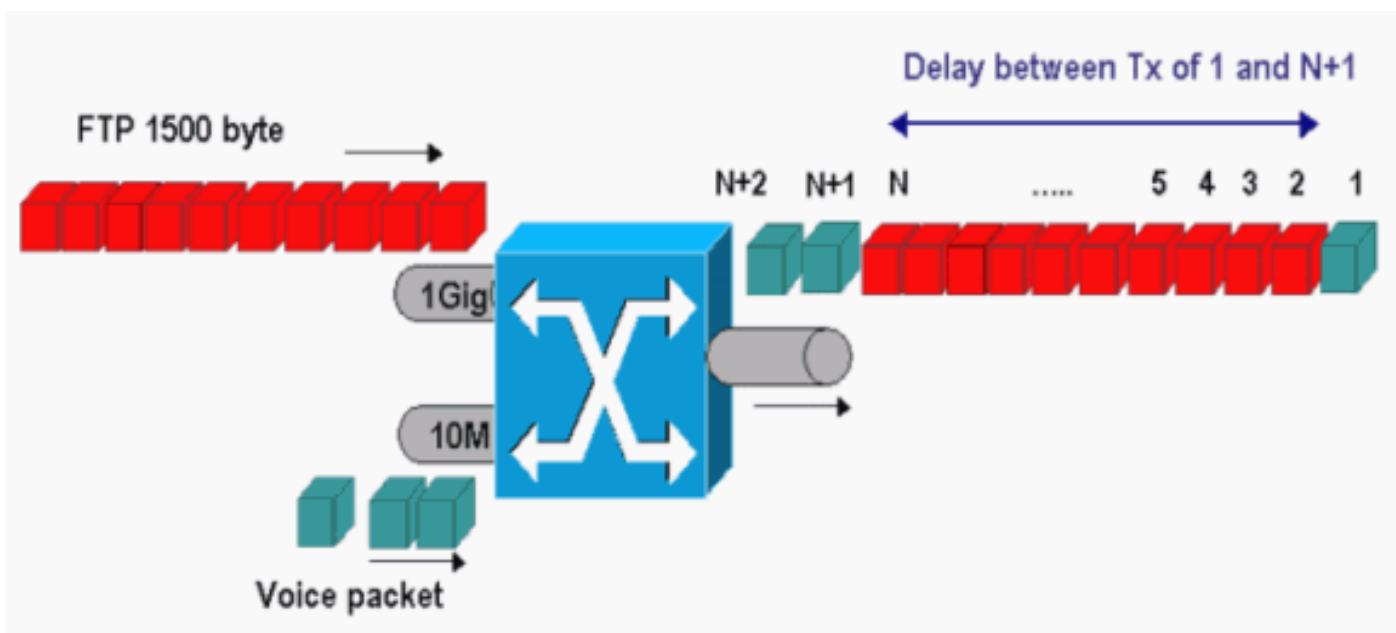
- $(10 \times 1500 \times 8) = 120.000$  bits transmitidos em 12 milissegundos

A implementação da programação de saída garante que os pacotes de voz com um CoS de 5 sejam colocados na fila de prioridade estrita e sejam enviados antes de qualquer pacote com um CoS inferior a 5. Isso reduz o atraso.

### Reduzir o atraso de sincronismo

Outro benefício importante do agendamento de saída é a redução do jitter. Jitter é a variação no atraso para pacotes dentro do mesmo fluxo. Este cenário de exemplo mostra como o agendamento de saída pode reduzir o jitter:





Neste cenário, a mesma porta de saída precisa enviar dois fluxos:

- Um fluxo de voz de entrada em uma porta Ethernet de 10 MB.
- Um fluxo FTP de entrada em uma porta de uplink 1 Gigabit Ethernet.

Os dois fluxos deixam o switch através da mesma porta de saída. Este exemplo mostra o que pode ocorrer sem o uso do agendamento de saída. Todos os pacotes de dados grandes podem ser intercalados entre dois pacotes de voz. Isso cria instabilidade na recepção do pacote de voz do mesmo fluxo. Há um atraso maior entre a recepção do pacote 1 e do pacote  $n + 1$  à medida que o switch transmite o pacote de dados grande. No entanto, o atraso entre  $n + 1$  e  $n + 2$  é negligenciável. Isso resulta em instabilidade no fluxo de tráfego de voz. Você pode facilmente evitar esse problema com o uso de uma fila de prioridade estrita. Assegure-se de mapear o valor de CoS dos pacotes de voz para a fila de prioridade estrita.

## Conclusão

Neste documento, você viu estudos de caso de como configurar e solucionar problemas de programação de fila de saída em um Catalyst 6500/6000 que executa o Cisco IOS Software. Você também viu as vantagens do agendamento de saída na maioria das redes com tráfego de voz:

- Evita a queda de tráfego crítico em caso de excesso de assinatura da porta de saída.
- Reduz o atraso.
- Reduz a tremulação.

## Informações Relacionadas

- [Programação da saída de QoS nos Switches da série Catalyst 6500/6000 executando o Software do sistema CatOS](#)
- [Entendendo a qualidade do serviço nos Switches da família Catalyst 6000](#)
- [Páginas de Suporte de Produtos de LAN](#)
- [Página de suporte da switching de LAN](#)
- [Suporte Técnico e Documentação - Cisco Systems](#)