

Compreenda e configurar o MDRR/WRED no Cisco 12000 Series Internet Router

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenções](#)

[Informações de Apoio](#)

[Visão geral de MDRR](#)

[Fila de prioridade do MDRR](#)

[Exemplo de MDRR](#)

[Suporte MDRR por tipo de mecanismo](#)

[Visão geral do WRED](#)

[Use a sintaxe de CoS existente para a configuração](#)

[Use o Modular QoS CLI \(MQC\) para a configuração](#)

[Comandos monitorar o Tratamento de Congestionamento e a vacância](#)

[O comando show interfaces](#)

[A mostra conecta o comando random do {number}](#)

[O entalhe do executivo \(y\) comando do {port} da fila do frfab do controlador da mostra](#)

[O entalhe do executivo \(y\) comando qm stat do frfab do controlador da mostra](#)

[Monitore o Tratamento de Congestionamento de entrada](#)

[O comando show interfaces](#)

[O comando show controller tofab queue do slot\(x\) do executivo](#)

[O comando da fila do tofab do controlador da mostra do slot\(x\) do executivo \(entalhe\) \(porta\)](#)

[O comando qm stat do tofab do controlador da mostra do slot\(x\) do executivo](#)

[Casos Práticos](#)

[Informações Relacionadas](#)

[Introdução](#)

Este documento avalia como configurar o gerenciamento de congestionamento do Cisco IOS® Software e os recursos para evitar congestionamento no Cisco 12000 Series Internet Router.

Depois que você lê este documento, você deve poder:

- Compreenda porque é importante configurar o Modified Deficit Round Robin (MDRR) e o Weighted Random Early Detection (WRED) em sua rede central.
- Compreenda a aplicação que é a base do MDRR e do WRED no Cisco 12000 Series.

- Configure MDRR e WRED usando a sintaxe de legado de CoS (Class of Service) e MQC (Modular QoS CLI).

Pré-requisitos

Requisitos

Os leitores deste documento devem estar cientes destes tópicos:

- Uma familiaridade geral com a arquitetura do Cisco 12000 Series Internet Router.
- Em particular, uma conscientização da arquitetura de enfileiramento e estes termos: O tofab (para a tela) – que descreve o lado de recepção se enfileira em uma placa de linha de entrada. O frfab (da tela) – que descreve o lado de transmissão se enfileira em uma placa de linha de partida.

Nota: Igualmente recomenda-se que você olha acima [como ler a saída do frfab do controlador da mostra | comandos de fila tofab em um Cisco 12000 Series Internet Router](#).

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- Todas as Plataformas do Cisco 12000, que incluem os 12008, 12012, 12016, 12404, 12406, 12410, e 12416.
- Cisco IOS Software Release 12.0(24)S1.

Nota: Embora as configurações neste documento sejam testadas no Cisco IOS Software Release 12.0(24)S1, todo o Cisco IOS Software Release que apoiar o Cisco 12000 Series Internet Router pode ser usado.

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a sua rede estiver ativa, certifique-se de que entende o impacto potencial de qualquer comando.

Convenções

Para obter mais informações sobre convenções de documento, consulte as [Convenções de dicas técnicas Cisco](#).

Informações de Apoio

Os métodos de enfileiramento definem o mecanismo de programação do pacote ou a ordem em que os pacotes de queue à relação para a transmissão no fio físico. Baseado na ordem e no número de vezes que uma fila está prestada serviços de manutenção por uma função do planejador, os métodos de enfileiramento igualmente apoiam garantias de largura de banda mínima e latências baixa.

É importante assegurar-se de que um mecanismo de programação do pacote apoie a arquitetura de switching em que é executado. O Weighted Fair Queuing (WFQ) é o algoritmo de

escalonamento conhecido para o alocamento de recursos em plataformas do Cisco Router com uma arquitetura com base em barramento. Contudo, não é apoiado no Cisco 12000 Series Internet Router. As filas de prioridade tradicionais e o Custom Queueing do Cisco IOS Software não são apoiados igualmente. Em lugar de, o Cisco 12000 Series usa um formulário especial de enfileirar o Modified Deficit Round Robin (MDRR) chamado, que fornece garantias da largura de banda relativa assim como uma fila de latência baixa. O M do MDRR representa “alterado”; adiciona a fila de prioridade comparada ao DRR onde nenhuma fila de prioridade esta presente. Para detalhes no MDRR, veja a seção de [visão geral de MDRR](#).

Além disso, o Cisco 12000 Series suporta WRED (Detecção antecipada aleatória ponderada) como política de queda dentro das filas MDRR. Este mecanismo de fuga de congestionamento fornece uma alternativa ao mecanismo da queda traseira do padrão. O congestionamento pode ser evitado por eliminações controladas.

A fuga de congestionamento e os mecanismos de gerenciamento tais como o WRED e o MDRR são particularmente importantes nas filas FrFab de interfaces externas relativamente de baixa velocidade, tais como as placas de linha separadas (LC). O Switch Fabric de alta velocidade pode entregar pacotes aos grupos de canais muito mais rapidamente do que os grupos de canais podem os transmitir. Enquanto se enfileirando/bufferes são controlados a nível da porta física, a pressão contrária em um canal pode impactar todos canais restantes nessa porta. Assim, é muito importante controlar essa pressão contrária com o WRED/MDRR, que limita ao impacto da pressão contrária aos canais na pergunta. Para detalhes sobre como controlar o sobrecarregamento de interface externa, não veja [pacotes ignorados do Troubleshooting e nenhuma gota da memória no Cisco 12000 Series Internet Router](#).

Visão geral de MDRR

Esta seção fornece uma vista geral do Modified Deficit Round Robin (MDRR).

Com o MDRR configurado como a estratégia de fila, as filas NON-vazias são servidas em sucessão, em um estilo round-robin. Cada vez que uma fila é servida, uma quantidade fixa de dados dequeued. O algoritmo presta serviços de manutenção então à fila seguinte. Quando uma fila é servida, o MDRR mantém-se a par do número de bytes de dados que dequeued além do valor configurado. Na passagem seguinte, quando a fila é servida outra vez, menos dados dequeued para compensar os dados excedentes que foram servidos previamente. Em consequência, o valor médio dos dados dequeued pela fila será próximo ao valor configurado. Além, o MDRR mantém uma fila de prioridade que obtenha servida em uma base preferencial. O MDRR é explicado em maiores detalhes nesta seção.

Cada fila dentro do MDRR é definida por duas variáveis:

- Valor quântico – Este é o número de bytes médio servido em cada um redondo.
- Contador de deficit – Isto é usado para seguir quantos bytes uma fila transmitiu em cada um redondo. É inicializado ao valor quântico.

Os pacotes em uma fila são servidos enquanto o contador de deficit é maior de zero. Cada pacote servido reduz o contador de déficit de um valor igual a seu comprimento em bytes. Uma fila não poderá mais ser atendida depois que o contador de déficit ficar zerado ou negativo. Em cada um o círculo novo, o contador de deficit de cada fila NON-vazia é aumentado por seu valor quântico.

Nota: Geralmente, o tamanho do quantum para uma fila não deve ser menor do que a unidade de transmissão máxima (MTU) da relação. Isto assegura-se de que o planificador serva sempre pelo

menos um pacote de cada fila NON-vazia.

Um peso relativo pode ser atribuído a cada fila de MDRR, com uma das filas no grupo definida como uma fila de prioridade. Os pesos atribuem a largura de banda relativa para cada fila quando a relação é congestionada. O algoritmo MDRR dequeues dados de cada fila em um estilo round-robin se há uns dados na fila a ser enviada.

Se todas as filas MDRR regulares possuírem dados, elas obterão serviços da seguinte maneira:

0-1-2-3-4-5-6-0-1-2-3-4-5-6...

Durante cada ciclo, uma fila pode dequeue um quantum baseado em seu peso configurado. Em placas de linha do motor 0 e do Engine 2, um valor de 1 é equivalente a dar à relação um o peso de seu MTU. Para cada incremento acima de 1, o peso da fila aumenta em 512 bytes. Por exemplo, se o MTU de uma determinada interface é 4470 e o peso de uma fila for configurado para 3, a cada tempo através da rotação, é permitido que $4470 + (3-1)*512 = 5494$ bytes sejam desenfileirados. Se duas filas normais DRR, Q0 e Q1, são, o Q0 configurado com um peso de 1 e o Q1 é configurado com um peso do 9. Se ambas as filas são congestionadas, cada vez com a rotação, Q0 seriam reservadas enviar 4470 bytes e o Q1 estaria permitido enviar $[4470 + (9-1)*512] = 8566$ bytes. Isto daria o tráfego que vai a Q0 aproximadamente 1/3 da largura de banda, e o tráfego que atravessa o Q1 aproximadamente 2/3 da largura de banda.

Nota: A fórmula padrão da de-fila usada para computar a atribuição da largura de banda de MDRR é $D = MTU + (weight-1)*512$. Nas versões mais cedo do que o Cisco IOS Software Release 12.0(21) S/ST, as placas de linha do motor 4 usaram um diferente dequeue a fórmula. A fim configurar o peso MDRR para uma atribuição de largura de banda correta, assegure-se de que você execute um Cisco IOS Software Release mais tarde de 12.0(21) S/ST.

Nota: A fórmula do quantum para as placas de linha do motor 4+ é (para a direção ToFab, esta é inválida para o frfab) $quantum = BaseWeight + \{BaseWeight * (QueueWeight - 1) * 512\}/MTU$. O BaseWeight é obtido com esta fórmula: $BaseWeight = \{() MTU + 512 - 1/512\} + 5$

Nota: Todos os cálculos são arredondados fora; isto é, todos os decimais são ignorados.

Nota: Para saber se uma placa de linha específica do motor apoia o MDRR, veja o [apoio MDRR pelo tipo de Engine](#).

Fila de prioridade do MDRR

O Cisco 12000 Series apoia um priority queue (PQ) dentro do MDRR. Esta fila fornece o baixos atraso e tremulação baixa exigidos pelo tráfego sensível ao tempo tal como a Voz sobre IP (VoIP).

Conforme observado acima, a Série Cisco 12000 não suporta o enfileiramento moderado ponderado (WFQ). Sendo assim, o PQ dentro de MDRR opera de maneira diferente do recurso de enfileiramento de baixa latência (LLQ) do software Cisco IOS, disponível para outras plataformas.

Uma diferença chave é como o programador de MDRR pode ser configurado para prestar serviços de manutenção ao PQ em um de dois modos, como catalogado na [tabela 1](#):

Tabela 1 – Como configurar o programador de MDRR para prestar serviços de manutenção ao

PQ em dois modos

	Alternate Mode (Modo Alternado)	Modo de prioridade estrita
Vantagens	Aqui, o PQ é prestado serviços de manutenção entre as outras filas. Ou seja o programador de MDRR presta serviços de manutenção alternativamente ao PQ e a todas as outras filas configuradas.	O PQ é prestado serviços de manutenção aqui sempre que está NON-vazio. Isto fornece o mais baixo atraso possível para o tráfego de correspondência.
Desvantagens	Este modo pode introduzir o tremor e atrasá-lo quando comparado ao modo de prioridade estrita.	Este modo pode morrer de fome outras filas, particularmente se os fluxos de harmonização são remetentes agressivos.

O modo alternativo pode exercitar menos controle no tremor e atrasá-lo. Se os começos do programador de MDRR para prestar serviços de manutenção a quadros do tamanho MTU de uns dados se enfileiram e então um pacote de voz chega no PQ, o planificador no modo alternativo serve completamente a fila sem prioridade até que seus alcances zero do contador de deficit. Durante este tempo, o PQ não é prestado serviços de manutenção, e os pacotes voip são atrasados.

Pelo contraste, no modo de prioridade estrita, os serviços de programador somente o pacote atual da NON-prioridade e comutam então ao PQ. O planificador começa prestar serviços de manutenção a uma fila sem prioridade somente depois que o PQ se torna completamente vazio.

É importante notar que a fila de prioridade no modo de prioridade alternativo está prestada serviços de manutenção mais de uma vez em um ciclo, e toma assim mais largura de banda do que outras filas com o mesmo peso nominal. Quanto mais é uma função de quantas filas são definidos. Por exemplo, com três filas, a fila de latência baixa é prestada serviços de manutenção duas vezes mais frequentemente que as outras filas, e envia duas vezes seu peso pelo ciclo. Se oito filas forem definidas, a fila de latência baixa será atendida sete vezes mais frequentemente e o peso efetivo será sete vezes maior. Assim, a largura de banda que a fila pode tomar é relacionada a como é servida frequentemente pelo arredondamento robin, que depende por sua vez de quantas filas são definidas em geral. No modo de prioridade alternativo, a fila de prioridade é configurada geralmente com um peso pequeno para esta razão particular.

Como um exemplo, supõe que quatro filas estão definidas: 0, 1, 2 e a fila de prioridade. No modo de prioridade alternativo, se todas as filas são congestionadas, seriam prestadas serviços de manutenção como segue: 0, llq, 1, llq, 2, llq, 0, llq, 1,..... em que llq significa fila de latência baixa.

Cada vez que uma fila é ocupada, ela pode enviar até o peso configurado para ela. Portanto, a largura de banda mínima que a fila de baixa latência pode ter é de:

- WL = peso da fila de latência baixa.
- W0, W1,... Wn = pesos das filas regulares DRR.
- n = número de filas regulares DRR usadas para esta relação.

- BW = largura de banda do link.

Para o modo de prioridade alternada, a largura de banda mínima da fila de latência mínima = $BW * n * WL / (n * WL + \text{Sum}(W0, Wn))$.

O peso é o único parâmetro variável no MDRR que pode ser configurado. Influencia o volume relativo de largura de banda que uma classe de tráfego pode usar, e quanto tráfego é enviado em uma volta. O uso de pesos maiores significa que o ciclo total toma mais por muito tempo, e aumenta possivelmente a latência.

Diretrizes de configuração

- É melhor configurar o peso da classe que tem o mais baixo requisito de largura de banda a 1 a fim manter o mais baixo possível o atraso e o tremor entre as outras classes.
- Selecione os valores de peso mais baixos possíveis. Comece com um peso de 1 para a classe com a menor largura de banda. Por exemplo, quando você usa duas classes com largura de banda de 50% para cada classe, você deve configurar 1 e 1. Não faz o sentido usar o 10 e o 10, porque não há nenhum impacto no desempenho quando você escolhe 1. Além disso, um peso maior apresenta mais tremulação.
- Um baixo valor do peso para o LLQ é muito importante, no modo alternativo para não adicionar especialmente demasiado atraso ou tremor às outras classes.

Exemplo de MDRR

O exemplo nesta seção é tomado do interior da *arquitetura de software de Cisco IOS@*, impressão Cisco.

Supõe que nós temos três filas:

- **A fila 0** – tem um quantum de 1500 bytes; É a fila de baixa latência, configurada para operar em modo alternado.
- **A fila 1** – tem um quantum de 3000 bytes.
- **A fila 2** – tem um quantum de 1500 bytes.

[Figura 1](#) ilustra o estado inicial das filas, junto com alguns pacotes que foram recebidos e enfileirados.

Figura 1 – Estado inicial MDRR

A fila 0 é prestada serviços de manutenção primeiramente, seu quantum é adicionada a seu contador de deficit, o pacote 1, que é 250 bytes, é transmitida, e seu tamanho é subtraído do contador de deficit. Porque o contador de deficit da fila 0 é ainda maior de 0 ($1500 - 250 = 1250$), o pacote 2 é transmitido também, e seu comprimento subtraído do contador de deficit. O contador de deficit da fila 0 é agora -250, assim que a fila 1 é prestada serviços de manutenção em seguida. [Figura 2](#) indica este estado.

Figura 2 – Estado subsequente MDRR

O contador de deficit da fila 1 é ajustado a 3000 ($0 + 3000 = 3000$), e os pacotes 4 e 5 é transmitido. Com cada pacote transmitido, subtraia o tamanho do pacote do contador de deficit, assim que o contador de deficit da fila 1 é reduzido a 0. [figuras 3](#) ilustra este estado.

Figura 3 – Estado MDRR quando o contador de deficit da fila 1 for zero

Nós precisamos de retornar do modo de prioridade alternativo à fila de serviços 0. Além disso, o

quantum é adicionado ao contador de deficit atual, e o contador de deficit da fila 0 é ajustado ao resultado (-250 + 1500 = 1250). O pacote 3 é transmitido agora porque o contador de déficit é maior que 0 e a fila 0 está vazia agora. [Quando uma fila é esvaziada, o contador de déficit é configurado para 0, como mostrado na Figura 4.](#)

Figura 4 – Estado MDRR quando uma fila for esvaziada

A fila 2 é prestada serviços de manutenção em seguida; seu contador de déficit está definido como 1500 (0 + 1500 = 1500). Os pacotes 7 a 10 são transmitidos, que deixa o contador de deficit em 500 (1500 - (4*250) = 500). Porque o contador de deficit é ainda maior de 0, o pacote 11 é transmitido igualmente.

[Quanto o pacote 11 é transmitido, a fila 2 está vazia e seu contador de déficit definido como 0, como mostra a Figura 5.](#)

Figura 5 – Estado MDRR quando o pacote 11 for transmitido

A fila 0 é atendida novamente em seguida (porque estamos no modo de prioridade alternada). Porque está vazia, nós a fila de serviços 1 em seguida, e transmitimos o pacote 6.

[Suporte MDRR por tipo de mecanismo](#)

O Cisco 12000 Series é compatível com cinco modelos de placas de linha com arquiteturas de Engine de encaminhamento da Camada 3 (L3). O apoio para o MDRR varia com o tipo do Engine de L3, e o tipo de cartão. Por exemplo, não há nenhum apoio para o MDRR e o WRED em placas de linha do motor 0 ATM. Use o comando show diag para determinar o tipo de L3 Engine das placas de linha instaladas.

```
router#show diag | include (SLOT | Engine) !--- The regular expression is case-sensitive. ...
SLOT 1 (RP/LC 1 ): 1 port ATM Over SONET OC12c/STM-4c Multi Mode L3 Engine: 0 - OC12 (622 Mbps)
SLOT 3 (RP/LC 3 ): 3 Port Gigabit Ethernet L3 Engine: 2 - Backbone OC48 (2.5 Gbps)
```

[MDRR em filas do tofab \(RX\)](#)

Você pode usar a “sintaxe de CoS existente” ou do “a interface de linha comando modular qos” para configurar o MDRR no Cisco 12000 Series. As seções mais recente neste documento discutem como configurar o MDRR com legado CoS ou QoS modular. Você deve configurar as filas do tofab com a sintaxe de CoS existente somente porque não apoiam a sintaxe mais nova do MQC. Veja a [tabela 2](#) para detalhes.

Tabela 2 – Detalhes no MDRR em filas do tofab (RX)

	Onde foram implementados	Tofab MDRR	Tofab PQ alternativo	ToFab Strict PQ	ToFab WRED
Eng0	Software	Não **	Não **	Sim	Sim
Eng1	-	Não	Não	Não	Não
Eng2	Hardware	Sim	Sim	Sim	Sim
Eng3	Hardware	Não	Sim	Sim	Sim
Eng4	Hardware	Sim	Sim	Sim	Sim

	e				
Eng4+	Hardware	Sim	Sim	Sim	Sim

** O MDRR é suportado no Mecanismo 0 LCs na direção ToFab (Rx), mas somente no modo prioridade estrita, não no modo prioridade alternativa. As sete filas restantes são suportadas normalmente.

As interfaces de entrada mantêm uma fila de saída virtual separada pelo destino LC. A maneira como essas filas são implementadas depende do tipo de Engine L3.

- Motor 0 – Os LC de entrada mantêm oito filas pelo slot de destino. Assim, o número máximo de filas é $16 \times 8 = 128$. Cada fila pode ser configurada separadamente.
- Engines 2, 3, 4 e 4+ – LCs de entrada mantêm oito filas por interface de destino. Com 16 slot de destino e 16 relações pelo entalhe, o número máximo de filas é $16 \times 16 \times 8 = 2048$. Todas as relações em um slot de destino devem usar os mesmos parâmetros.

[MDRR em Filas FrFab \(Tx\)](#)

O MDRR nas filas FrFab opera-se consistentemente se executado no hardware ou no software. Todos os tipos do Engine de L3 apoiam oito filas de classe para cada interface externa. Veja a [tabela 3](#) para detalhes.

Tabela 3 – Detalhes no MDRR em filas do frfab (Tx)

	Onde foram implementados	PQ alternativo de FrFab	FrFab PQ restrito	FrFab WRED
Eng0	Software ¹	Não	Sim	Sim
Eng1	-	Não	Não	Não
Eng2	Hardware	Sim ²	Sim	Sim
Eng3	Hardware	Não	Sim	Sim
Eng4	Hardware	Sim	Sim	Sim
Eng4+	Hardware	Sim	Sim	Sim

¹Support para o MDRR em filas FrFab do motor 0 LC é introduzido nestas versões de Cisco IOS Software:

- Cisco IOS Software Release 12.0(10)S - 4xOC3 e 1xOC12 POS, 4xOC3, e 1xCHOC12/STM4.
- Cisco IOS Software Release 12.0(15)S - 6xE3 e 12xE3.
- Cisco IOS Software Release 12.0(17)S - 2xCHOC3/STM1.

²You deve configurar o MDRR alternativo na direção de FrFab com a sintaxe de CoS existente.

Nota: Os apoios MDRR 3xGE LC nas filas do tofab e, como do Cisco IOS Software Release 12.0(15)S, nas filas FrFab com duas limitações, a saber, um quantum fixo, e uma única fila de CoS para cada relação. A fila de prioridade apoia um quantum que possa ser configurado, e modos de prioridade restritos e alternativos. Todas as três interfaces compartilham um único PQ.

Nota: Veja os Release Note dos Cisco 12000 Series Router para as últimas informações sobre das características de QoS apoiadas no Cisco 12000 Series LC.

Visão geral do WRED

A WRED (Detecção antecipada aleatória ponderada) foi desenvolvida para impedir os efeitos nocivos da congestão da interface no throughput da rede.

Figura 6 – Probabilidade de queda de pacote de informação WRED

Veja o [Weighted Random Early Detection no Cisco 12000 Series Router](#) para uma explicação dos parâmetros de WRED. O mínimo, o máximo, e os parâmetros da probabilidade da marca descrevem a curva real do Random Early Detection (RED). Quando a média tornada mais pesada da fila está abaixo do limiar mínimo, nenhum pacote está deixado cair. Quando a média tornada mais pesada da fila está acima do ponto inicial da fila máxima, todos os pacotes estão deixados cair até as gotas médias abaixo do limiar máximo. Quando a média está entre o mínimo e os limiares máximos, a probabilidade que o pacote estará deixado cair pode ser calculada por uma linha reta do limiar mínimo (a probabilidade da gota será 0) ao limiar máximo (a probabilidade da gota é igual ao denominador de probabilidade 1/mark).

A diferença entre o VERMELHO e o WRED é que o WRED pode seletivamente rejeitar o tráfego de prioridade mais baixa quando a relação começa a obter congestionada, e pode fornecer características de desempenho diferenciadas para classes de serviço (COS) diferentes. À revelia, o WRED usa um perfil VERMELHO diferente para cada peso (Precedência IP - 8 perfis). Ele desconecta pacotes menos importantes de forma mais agressiva do que pacotes mais importantes.

É um desafio complexo para ajustar parâmetros de WRED para controlar a profundidade de fila, e depende de muitos fatores, que incluem:

- Carga de tráfego e perfil oferecidos.
- Relação da carga à capacidade disponível.
- Comportamento do tráfego na presença da congestão.

Estes fatores variam a rede pela rede e, por sua vez, dependem dos serviços oferecidos e dos clientes que usam aqueles serviços. Assim, nós não podemos fazer as recomendações que se aplicam aos ambientes de cliente específicos. Contudo, a [tabela 4](#) descreve geralmente os valores recomendados baseados na largura de banda do link. Nesse caso, não diferenciamos as características da queda segundo as diferentes classes de serviço.

Tabela 4 – Valores recomendados baseados na largura de banda do link

Largura de banda	Largura de Banda Teórica (kbps)	BW1 Físico (kbps)	Limiar mínimo	Limiar Máximo
OC3	155000	149760	94	606
OC12	622000	599040	375	2423
OC48	2400000	239616	1498	9690
OC192	10000000	9584640	5991	38759

Existem várias restrições que são levadas em conta quando se calculam os valores limiares acima. Por exemplo, o máximo da utilização do enlace ao minimizar a profundidade média da fila, a diferença entre o máximo e o mínimo devem ser uma potência de dois (devido à limitação do hardware). Com base na experiência e na simulação, a profundidade instantânea máxima de uma filha controlada por RED é menor que 2 MaxTh. Para o OC48 ou maior, 1 MaxTh e assim por diante. No entanto, a determinação exata desses valores está além do escopo deste documento.

Nota: O valor da constante de ponderação exponencial não precisa de ser configurado no Engine 2 e acima das placas de linha, desde que a amostra da fila de hardware é usada pelo contrário. Para o motor 0 LC, estes valores podem ser configurados:

- DS3 – 9
- oc3 – 10
- oc12 – 12

Nota: Não há suporte para WRED nos LCs do Mecanismo 1.

Enquanto as seguintes seções explicam, você pode usar a sintaxe de CoS existente e a sintaxe de MQC mais nova para configurar o WRED.

[Use a sintaxe de CoS existente para a configuração](#)

A sintaxe de Classe de serviço (CoS) herdada do Cisco 12000 Series utiliza um modelo cos-queue-group para definir um conjunto de definições de CoS. Você aplica então o molde ao tofab e filas FrFab em de entrada ou em interfaces externas, respectivamente.

[Passo 1: Defina um cos-queue-group](#)

O comando **cos-queue-group** cria um molde Nomeado do MDRR e dos parâmetros de WRED. Estão aqui os parâmetros de configuração disponíveis no CLI:

```
Router(config)#cos-queue-group oc12 Router(config-cos-que)#? Static cos queue commands: default
Set a command to its defaults dscp Set per DSCP parameters, Engine 3 only exit Exit from COS
queue group configuration mode exponential-weighting-constant Set group's RED exponential weight
constant. (Not used by engine 0, 1 or 2 line cards) no Negate a command or set its defaults
precedence Set per precedence parameters queue Set individual queue parmeters random-detect-
label Set RED drop criteria traffic-shape Enable Traffic Shaping on a COS queue group
```

Com MDRR, você pode traçar a Precedência IP às filas MDRR e configurar a fila de latência baixa especial. Você pode usar o parâmetro de precedência sob o **comando cos-queue-group** para este:

```
precedence <0-7> queue [ <0-6> | low-latency]
```

Você pode mapear uma precedência de IP particular para uma fila MDRR comum (fila de 0 a 6) ou pode mapeá-la para uma fila de prioridade. O comando acima permite que você mapeie diversas precedências de IP na mesma fila.

Nota: Recomenda-se que você usa a precedência 5 para a fila de latência baixa. A Precedência 6 é usada para atualizações de roteamento.

Você pode dar a cada fila MDRR um o peso relativo, com uma das filas no grupo definido como uma fila de prioridade. Você pode usar o **comando queue** sob o **cos-queue-group** fazer isto:

```
queue <0-6> <1-2048> queue low-latency [alternate-priority | strict-priority] <1-2048> !--- The weight option is not available with strict priority.
```

Use o comando `cos-queue-group` para definir os parâmetros da WRED:

```
random-detect-label <label> <minimum-threshold> <maximum-threshold> <mark-probability denominator>
```

Está aqui um exemplo de um **cos-queue-group** nomeado **oc12**. Define três classes de tráfego: fila 0, 1, e latência baixa. Traça os valores de precedência IP 0 - 3 para enfileirar 0, valores de precedência 4, 6, e 7 para enfileirar 1, e precedência 5 à fila de latência baixa. Mais largura de banda foi atribuída à fila 1.

Exemplo de configuração

```
cos-queue-group oc12
!--- Creation of cos-queue-group called "oc12".
precedence 0 queue 0 !--- Map precedence 0 to queue 0.
precedence 0 random-detect-label 0 !--- Use RED profile
0 on queue 0. precedence 1 queue 0 precedence 1 random-
detect-label 0 precedence 2 queue 0 precedence 2 random-
detect-label 0 precedence 3 queue 0 precedence 3 random-
detect-label 0 !--- Precedence 1, 2 and 3 also go into
queue 0. precedence 4 queue 1 precedence 4 random-
detect-label 1 precedence 6 queue 1 precedence 6 random-
detect-label 1 precedence 7 queue 1 precedence 7 random-
detect-label 1 precedence 5 queue low-latency !--- Map
precedence 5 to special low latency queue. !--- We do
not intend to drop any traffic from the LLQ. We have an
SLA !--- that commits not to drop on this queue. You
want to give it all !--- the bandwidth it requires.
Random-detect-label 0 375 2423 1 !--- Minimum threshold
375 packets, maximum threshold 2423 packets. !--- Drop
probability at maximum threshold is 1. random-detect-
label 1 375 2423 1 queue 1 20 !--- Queue 1 gets MDRR
weight of 20, thus gets more Bandwidth. queue low-
latency strict-priority !--- Low latency queue runs in
strict priority mode.
```

[Etapa 2 - Criar um slot-table-cos para as filas ToFab](#)

Para evitar a cabeça da linha que obstrui, as interfaces de entrada no Cisco 12000 Series mantêm uma fila de saída virtual pelo slot de destino. Vá a uma placa de linha usando o comando `attach` e execute `executar-no comando show controller tofab queue` (ou incorpore diretamente o `execute-on slot 0 comandos show controllers tofab queue`) ver estas filas de saída virtuais. O exemplo de saída capturado diretamente do console LC é fornecido abaixo. Veja [como ler a saída do frfab do controlador da mostra | comandos de fila tofab em um Cisco 12000 Series Internet Router](#).

```
LC-Slot1#show controllers tofab queues Carve information for ToFab buffers SDRAM size: 33554432
bytes, address: 30000000, carve base: 30029100 33386240 bytes carve size, 4 SDRAM bank(s), 8192
bytes SDRAM pagesize, 2 carve(s) max buffer data size 9248 bytes, min buffer data size 80 bytes
40606/40606 buffers specified/carved 33249088/33249088 bytes sum buffer sizes specified/carved
Qnum Head Tail #Qelem LenThresh ---- ---- ----- ----- 5 non-IPC free queues:
20254/20254 (buffers specified/carved), 49.87%, 80 byte data size 1 17297 17296 20254 65535
12152/12152 (buffers specified/carved), 29.92%, 608 byte data size 2 20548 20547 12152 65535
6076/6076 (buffers specified/carved), 14.96%, 1568 byte data size 3 32507 38582 6076 65535
1215/1215 (buffers specified/carved), 2.99%, 4544 byte data size 4 38583 39797 1215 65535
809/809 (buffers specified/carved), 1.99%, 9248 byte data size 5 39798 40606 809 65535 IPC
```

```
Queue: 100/100 (buffers specified/carved), 0.24%, 4112 byte data size 30 72 71 100 65535 Raw
Queue: 31 0 17302 0 65535 ToFab Queues: Dest Slot 0 0 0 0 65535 1 0 0 0 65535 2 0 0 0 65535 3 0
0 0 65535 4 0 0 0 65535 5 0 17282 0 65535 6 0 0 0 65535 7 0 75 0 65535 8 0 0 0 65535 9 0 0 0
65535 10 0 0 0 65535 11 0 0 0 65535 12 0 0 0 65535 13 0 0 0 65535 14 0 0 0 65535 15 0 0 0 65535
Multicast 0 0 0 65535 LC-Slot1#
```

Use o comando **slot-table-cos** traçar um **cos-queue-group** Nomeado a uma fila de saída virtual do destino. Você pode configurar um molde original do **cos-queue-group** para cada fila de saída

```
Router(config)#slot-table-cos table1 Router(config-slot-cos)#destination-slot ? <0-15>
Destination slot number all Configure for all destination slots Router(config-slot-
cos)#destination-slot 0 oc48 Router(config-slot-cos)#destination-slot 1 oc48 Router(config-slot-
cos)#destination-slot 2 oc48 Router(config-slot-cos)#destination-slot 3 oc48 Router(config-slot-
cos)#destination-slot 4 oc12 Router(config-slot-cos)#destination-slot 5 oc48 Router(config-slot-
cos)#destination-slot 6 oc48 Router(config-slot-cos)#destination-slot 9 oc3 Router(config-slot-
cos)#destination-slot 15 oc48
```

Nota: A configuração acima usa três moldes, oc48 Nomeado, oc12, e oc3. A configuração para o **cos-queue-group** nomeado oc12 é segundo as indicações de Step1. Similarmente, configurar oc3 e oc48. Recomenda-se que você aplica um template exclusivo a um grupo de relações baseadas na largura de banda e no aplicativo.

[Etapa 3 - Aplique um slot-table-cos a uma interface de entrada](#)

Use o comando **rx-cos-slot** para aplicar um **slot-table-cos** a um LC.

```
Router(config)#rx-cos-slot 0 ? WORD Name of slot-table-cos Router(config)#rx-cos-slot 0 table1
Router(config)#rx-cos-slot 2 table1
```

[Etapa 4 - Aplique um cos-queue-group a uma interface externa](#)

O Cisco 12000 Series mantém uma fila separada pela interface externa. Para ver estas filas, anexe à placa de linha CLI. Use o comando **attach**, e execute então o comando **show controller frfab queue**, como ilustrado aqui:

```
LC-Slot1#show controller frfab queue ===== Line Card (Slot 2) ===== Carve information for
FrFab buffers SDRAM size: 16777216 bytes, address: 20000000, carve base: 2002D100 16592640 bytes
carve size, 0 SDRAM bank(s), 0 bytes SDRAM pagesize, 2 carve(s) max buffer data size 9248 bytes,
min buffer data size 80 bytes 20052/20052 buffers specified/carved 16581552/16581552 bytes sum
buffer sizes specified/carved Qnum Head Tail #Qelem LenThresh ---- ---- ---- ----- 5
non-IPC free queues: 9977/9977 (buffers specified/carved), 49.75%, 80 byte data size 1 101 10077
9977 65535 5986/5986 (buffers specified/carved), 29.85%, 608 byte data size 2 10078 16063 5986
65535 2993/2993 (buffers specified/carved), 14.92%, 1568 byte data size 3 16064 19056 2993 65535
598/598 (buffers specified/carved), 2.98%, 4544 byte data size 4 19057 19654 598 65535 398/398
(buffers specified/carved), 1.98%, 9248 byte data size 5 19655 20052 398 65535 IPC Queue:
100/100 (buffers specified/carved), 0.49%, 4112 byte data size 30 77 76 100 65535 Raw Queue: 31
0 82 0 65535 Interface Queues: 0 0 0 0 65535 1 0 0 0 65535 2 0 0 0 65535 3 0 0 0 65535
```

Use o comando **tx-cos** aplicar um molde do **cos-queue-group** a uma fila de interface. Como mostrado aqui, você aplica o conjunto de parâmetro diretamente à relação; nenhuma tabela é precisada. Neste exemplo, *pos48* é o nome de um conjunto de parâmetro.

```
Router(config)#interface POS 4/0 Router(config-if)#tx-cos ? WORD Name of cos-queue-group
Router(config-if)#tx-cos pos48
```

Use o comando **show cos** para confirmar sua configuração:

```
Router#show cos !--- Only some of the fields are visible if MDRR is configured on Inbound !---
or Outbound interfaces. Interface Queue cos Group Gi4/0 eng2-frfab !--- TX-cos has been applied.
Rx Slot Slot Table 4 table1 !--- rx-cos-slot has been applied. Slot Table Name - table1 1 eng0-
tofab 3 eng0-tofab !--- slot-table-cos has been defined. cos Queue Group - eng2-tofab !--- cos-
queue-group has been defined. Prec Red Label [min, max, prob] Drr Queue [deficit] 0 0 [6000,
```

```
15000, 1/1] 0 [10] 1 1 [10000, 20000, 1/1] 1 [40] 2 1 [10000, 20000, 1/1] 1 [40] 3 1 [10000,
20000, 1/1] 0 [10] 4 2 [15000, 25000, 1/1] 2 [80] 5 2 [15000, 25000, 1/1] 2 [80] 6 no drop low
latency 7 no drop low latency
```

Nota: O legado CLI igualmente usa a sintaxe de precedência para o tráfego do Multiprotocol Label Switching (MPLS). O roteador trata os bit MPLS como se são bit do tipo do IP de serviço (ToS) e põe os pacotes apropriados nas filas corretas. Isso não é totalmente verdadeiro para o MQC. O MPLS QoS é fora do âmbito deste documento.

Use o Modular QoS CLI (MQC) para a configuração

O objetivo do Modular QoS CLI (MQC) de Cisco é conectar todas as características de QoS diferentes em uma maneira lógica, a fim simplificar a configuração de características do Qualidade de Serviço (QoS) do Cisco IOS Software. Por exemplo, a classificação é feita separadamente do enfileiramento, do policiamento, e de dar forma. Fornece uma única estrutura da configuração para QoS que molde-é baseado. Estão aqui alguns pontos a recordar sobre a configuração de MQC:

- Pode facilmente ser aplicada a e removido de uma relação.
- Pode facilmente ser reutilizada (a mesma política pode ser aplicada às interfaces múltiplas).
- Oferece uma única estrutura da configuração para QoS que o permite de provision, monitorar, e pesquisar defeitos facilmente.
- Fornece um de mais alto nível da abstração.
- É plataforma independente.

No Cisco 12000 Series, os comandos mqc podem ser usados em vez da sintaxe do Classe de serviço (CoS) do legado.

O apoio MQC no Cisco 12000 Series não implica que o mesmo disponível ajustado da característica de QoS em uma outra plataforma, tal como o Cisco 7500 Series, está agora disponível no Cisco 12000. O MQC fornece uma sintaxe comum em que um comando conduz a uma função ou a um comportamento compartilhado. Por exemplo, o **comando bandwidth** executa uma garantia de largura de banda mínima. O Cisco 12000 Series usa o MDRR como o mecanismo de programação para fazer a reserva de largura de banda, quando o Cisco 7500 Series usar o WFQ. O algoritmo do principal complementa a plataforma particular.

Importante, somente as filas FrFab apoiam a configuração das características de QoS com o MQC. Porque as filas do tofab são filas de saída virtuais, e filas de entrada não verdadeiras, não são apoiadas pelo MQC. Devem ser configurados com comandos de CoS do legado.

Apoio das lista da [tabela 5](#) para o MQC pelo tipo do Engine de L3.

Tabela 5 – Apoio para o MQC para tipos do Engine de L3

Tipo de L3 Engine	Mecanismo 0	Mecanismo 1	Mecanismo 2	Mecanismo 3	Mecanismo 4	Motor 4+
Suporte a MQC	Sim	Não	Sim	Sim	Sim	Sim
Versão do IOS	12.0(15)S	-	12.0(15)S ¹	12.0(21)S	12.0(22)S	12.0(22)S

¹Remember estas exceções com apoio MQC nas placas de linha do motor 0 e 2 (LC) s:

- 2xCHOC3/STM1 - Introduzido na versão 12.0(17)S.
- 1xOC48 DPT introduzido na versão 12.0(18)S.
- 8xOC3 ATM Planejado para 12.0(22)S.

O MQC usa estas três etapas para criar uma política de QoS:

1. Defina uma ou mais classes de tráfego com o comando class-map.
2. Crie uma política de QoS com o comando policy-map e atribua ações de QoS, como largura de banda ou prioridade, a uma classe de tráfego nomeada.
3. Use o **comando service-policy** anexar um mapa de política à fila FrFab de uma interface externa.

Use o **comando show policy-map interface** monitorar sua política.

Veja a [vista geral da interface da linha de comando da Qualidade de Serviço modular](#) para mais informações.

Etapa 1 - Defina mapas de classe

O comando **class-map** é usado definir classes de tráfego. Internamente, no Cisco 12000 Series, o comando **class-map** atribui uma classe a uma fila específica de CoS na placa de linha (veja [etapa 4](#) para detalhes).

O comando **class-map** apoia “compatível com qualquer”, que coloca os pacotes que combinam algumas das instruções compatível na classe, e “compatível com todos”, que coloca pacotes nesta classe somente quando todas as indicações são verdadeiras. Estes comandos criam uma classe nomeada "Prec_5", e classificam todos os pacotes com uma Precedência IP de 5 a esta classe:

```
Router(config-cmap)#match ? access-group Access group any Any packets class-map Class map
destination-address Destination address fr-dlci Match on fr-dlci input-interface Select an input
interface to match ip IP specific values mpls Multi Protocol Label Switching specific values not
Negate this match result protocol Protocol qos-group Qos-group source-address Source address
Router(config-cmap)#match ip precedence 5
```

[Apresente 6](#) lista os critérios de verificação de repetição de dados apoiados para cada tipo do Engine de L3.

Tabela 6 – Critérios de verificação de repetição de dados apoiados para os Engine de L3

	Motor 0, 2	Mecanismo 3	Mecanismo 4	Motor 4+
Precedência IP	Sim	Sim	Sim	Sim ¹
acesso-grupo	Não	Sim	Não	Não
mpls exp	Não	Sim	Não	Sim (12.0.26S)
dscp IP	Não	Sim	Não	Sim (12.0.26S)
qos-grupo	Não	Sim	Não	Não
x/y da	Não	Sim	Não	Não

interface de entrada POS do fósforo		(somente como política de recepção)		
-------------------------------------	--	-------------------------------------	--	--

¹ ingresso/saída desde 12.0.26S

[Etapa 2 - Crie um mapa de política](#)

O comando **policy-map** é usado atribuir políticas de manejo de pacote ou ações a umas ou várias classes definidas. Por exemplo, quando você atribui uma reserva de largura de banda, ou aplique um perfil aleatório da gota.

O Cisco 12000 Series apoia um subconjunto de características MQC, com base na arquitetura de alta velocidade dos Engine de L3. [Apresente 7](#) alista os comandos que são apoiados:

Tabela 7 – Comandos suportados

Comando	Descrição
largura de banda	Fornecer uma garantia de largura de banda mínima durante períodos de congestionamento. Especifica-se como uma porcentagem da velocidade do link ou como um valor absoluto. Se uma classe não usa nem precisa a largura de banda igual aos kbps reservados, a largura de banda disponível pode ser usada por outras classes de largura de banda.
police, shape	Limita a quantidade de tráfego que uma classe pode transmitir. Estes comandos são levemente diferentes na função. O comando police identifica o tráfego que excede a largura de banda configurada, e deixa-o cair ou observa-. Os buffers de comando shape todo o tráfego excedente e programam-no para a transmissão em uma taxa constante, mas não o deixam cair nem observam-no.
Limite de fila	Atribui um comprimento fixo à fila de uma determinada classe de tráfego. Você pode especificar este em número dos pacotes que podem ser realizados na fila.
prioridade	Identifica uma fila como uma fila de latência baixa. MQC suporta modo restrito somente para um PQ. O modo alternativo não é apoiado com o MQC. Use o comando priority sem um valor da porcentagem permitir o modo de prioridade estrita. Nota: A aplicação do comando priority no Cisco 12000 Series difere da aplicação no outro Roteadores que executa o Cisco IOS Software. Nesta plataforma, o tráfego de

	prioridade não é limitado ao valor configurado dos kbps durante períodos de congestionamento. Assim, você deve igualmente configurar o comando police limitar quanto largura de banda uma classe de prioridade pode usar e assegurar a largura de banda adequada para outras classes. Neste tempo, o comando police é apoiado somente em placas de linha do Engine 3. Nas outras placas de linha do motor, somente o class-default é permitido quando você configura uma classe de prioridade.
aleatório-detecte	Atribui um perfil WRED. Use o comando random-detect precedence configurar valores não-padrão WRED pelo valor de precedência IP.

No Engine 3 LC, você deve configurar as filas FrFab com o Modular QoS CLI (MQC); a interface de linha do comando legacy (CLI) não é apoiada.

Quando você configura o **comando bandwidth**, note que os LC do motor 0 e 2 apoiam seis classes de largura de banda somente. Uma sétima classe pode ser usada para o serviço da latência baixa e uma oitava classe, que seja class-default, é usada para todo o tráfego deharmonização. Portanto, você tem um total de oito filas. O padrão da classe não é utilizado como classe de prioridade.

No Engine 3 LC, o **comando bandwidth percent** é traduzido em um valor dos kbps, que varie com a taxa de enlace subjacente, e configurado então diretamente na fila. A precisão desta garantia de largura de banda mínima é 64 kbps.

Embora nenhuma conversão a um valor quântico seja feita com o **comando bandwidth**, todas as filas têm um quantum. No Engine 3 LC, o valor quântico é ajustado baseado internamente na unidade de transmissão máxima (MTU) da relação, e ajustado igualmente para todas as filas. Não existe mecanismo CLI MQC para modificar esse valor quântico, direta ou indiretamente. O valor quântico deve ser superior ou igual à interface MTU. Internamente, o valor quântico está nas unidades de 512 bytes. Assim, com um MTU de 4470 bytes, o valor quântico mínimo do MTU deve ser 9.

[MDRR no Engine 3 LC](#)

Esta seção fornece notas de configuração para executar o WRED e o MDRR no Engine 3 LC.

- A largura de banda de MDRR configurada no CLI é traduzida a uma quantidade que corresponde ao L2 (por exemplo, as despesas gerais L1 são removidas). Em seguida, essa quantidade é arredondada para os próximos 64 kbps e programada no hardware.
- Três perfis WRED diferentes são suportados para uma classe.
- O WRED (limiar máximo - limiar mínimo) é aproximado à potência a mais próxima de 2. O limiar mínimo está ajustado então automaticamente quando o limiar máximo for mantido inalterado.
- Há suporte para o valor de probabilidade Mark 1.

- A configuração constante de ponderação exponencial não é apoiada.
- A Precedência IP, os bit MPLS EXP, e os valores DSCP são apoiados.

Nota: Cada porta ou canal nas placas de linha Frostbite Tetra (4GE-SFP-LC=) ou CHOC12/DS1-IR-SC= têm quatro filas atribuídas à revelia. As quatro filas consistem no seguinte:

- Uma classe da fila de prioridade (LLQ)
- Uma classe da fila padrão
- Duas classes normais da NON-prioridade

Ao aplicar uma serviço-política que contém mais do que estas quatro classes (1 HPQ, 2 LPQs e class-default) à relação, o seguinte erro será relatado:

A #service-política do roteador (config-if) output a MDRR-política

% não bastante recursos do Enfileiramento disponíveis para satisfazer o pedido.

Até à data de 12.0(26)S, um comando foi adicionado para a placa de linha 4GE-SFP-LC= Tetra que permite a configuração de oito queues/VLAN em vez de quatro. As oito filas consistem no seguinte:

- Um LLQ
- Uma fila do class-default
- Seis filas normais

O uso deste comando exigirá um reload do microcódigo da placa de linha e conduzirá à capacidade para configurar somente 508 VLAN em vez de 1022. A sintaxe do comando é:

filas de interface 8 dos qos do <slot-> do hw-module slot do [no]

Por exemplo:

filas de interface 8 dos qos do entalhe 2 do Router(config)#hw-módulo

aviso: Por favor micro recarregamento a placa de linha para que este comando tome o efeito

Reload 2 de Router(config)#microcode

Este comando estará disponível para a placa de linha Frostbite CHOC12/DS1-IR-SC= em 12.0(32)S

Exemplo #1 - comando bandwidth percent

Este exemplo aloca 20% de largura de banda para class Prec_4 traffic e 30% para traffic of class Prec_3 traffic. Deixa os por cento restantes dos 50 pés à classe de padrão classe.

Além disso, ele configura a WRED como o mecanismo de queda em todas as classes de dados.

Exemplo #1 - percentagem de largura de banda

```

policy-map GSR_EXAMPLE
  class Prec_4
    bandwidth percent 20
    random-detect
    random-detect precedence 4 1498 packets 9690 packets 1
  !--- All data classes should have WRED configured. class

```

```
Prec_3 bandwidth percent 30 random-detect random-detect
precedence 3 1498 packets 9690 packets 1 class class-
default !--- Class-default uses any leftover bandwidth.
random-detect random-detect precedence 2 1498 packets
9690 packets 1 random-detect precedence 1 1498 packets
9690 packets 1 random-detect precedence 0 1498 packets
9690 packets 1
```

Exemplo #2 - comando bandwidth {kbps}

Este exemplo ilustra como aplicar o comando bandwidth como um valor absoluto dos kbps em vez de uma porcentagem.

Exemplo #2 - largura de banda {kbps}

```
policy-map GSR_EXAMPLE
  class Prec_4
    bandwidth 40000
    !--- Configures a minimum bandwidth guarantee of 40000
kbps or 40 Mbps in !--- times of congestion. Random-
detect random-detect precedence 4 1498 packets 9690
packets 1 class Prec_3 bandwidth 80000 !--- Configures a
minimum bandwidth guarantee of 80000 kbps or 80 Mbps in
!--- times of congestion. Random-detect random-detect
precedence 3 1498 packets 9690 packets 1 class class-
default !--- Any remaining bandwidth is given to class-
default. Random-detect random-detect precedence 2 1498
packets 9690 packets 1 random-detect precedence 1 1498
packets 9690 packets 1 random-detect precedence 0 1498
packets 9690 packets 1
```

Exemplo 3 - comando priority

Este exemplo é projetado para os provedores de serviços que usam o Cisco 12000 Series Router como um roteador da ponta de provedor MPLS (PE) e o precisam de configurar uma política de serviços de QoS no link entre o roteador de PE e o roteador do edge de cliente (CE). Coloca pacotes da Precedência IP 5 em uma fila de prioridade, e limita a saída dessa fila ao 64 Mbps. Atribui então uma parcela da largura de banda remanescente às classes de largura de banda.

Todas as filas de classe da NON-prioridade são configuradas com o comando **random-detect** permitir o WRED como a política da gota. Toda a classe de largura de banda e class-default devem ter o WRED configurado explicitamente.

Exemplo #3 - prioridade

```
policy-map foo
  class Prec_5
    police 64000000 conform-action transmit exceed-
action drop
    !--- The police command is supported on Engine 3 line
cards. priority class Prec_4 bandwidth percent 30
random-detect random-detect precedence 4 1498 packets
9690 packets 1 class Prec_3 bandwidth percent 10 random-
detect random-detect precedence 3 1498 packets 9690
packets 1 class Prec_2 bandwidth percent 10 random-
detect random-detect precedence 2 1498 packets 9690
packets 1 class Prec_1 bandwidth percent 10 random-
detect random-detect precedence 1 1498 packets 9690
packets 1 class Prec_0 bandwidth percent 25 random-
detect random-detect precedence 0 1498 packets 9690
```

```
packets 1 class class-default random-detect random-
detect precedence 6 1498 packets 9690 packets 1 random-
detect precedence 7 1498 packets 9690 packets 1
```

Etapa 3 - Atribua um mapa de política a uma fila da interface externa

Como mencionado acima, o MQC trabalha somente com as filas FrFab em uma interface externa. Para aplicar um mapa de política definido, use o **comando service-policy output**, como mostrado aqui:

```
Router(config)#interface POS 0/0 Router(config-if)#service-policy ? history Keep history of QoS
metrics input Assign policy-map to the input of an interface output Assign policy-map to the
output of an interface Router(config-if)#service-policy output ? WORD policy-map name
Router(config-if)#service-policy output GSR_EXAMPLE
```

Etapa 4 - Monitore e verifique a política de serviços

Use o **comando show policy-map interface** ver o aplicativo de uma política. O **comando show policy-map interface** indica o seguinte:

- Largura de banda configurada e classes de prioridade e os critérios em correspondência.
- Alguns perfis WRED.
- Parâmetros da forma e da política.
- Contabilidade e taxas do tráfego.
- A fila interna de CoS a que uma classe particular é traçada. Estas filas são providas pelo mesmo deslocamento predeterminado que é usado na saída do **comando show controller frfab queue**.

Estão aqui um exemplo de uma configuração completa e os **comandos show** monitorar a política:

Configuração completa

```
class-map match-all class1
  match ip precedence 1
class-map match-all class2
  match ip precedence 2
!--- Step 1 - Configure traffic classes. ! policy-map
policy Class class1 bandwidth percent 10 random-detect
random-detect precedence 1 375 packets 2423 packets 1
Class class2 bandwidth percent 20 random-detect !---
Step 2 - Configure a policy-map. ! interface POS6/0 ip
address 12.1.1.1 255.255.255.0 no ip directed-broadcast
no keepalive service-policy output policy !--- Step 3-
Attach policy-map to the interface.
```

Use o **comando show policy-map interface** ver a política configurada na relação, junto com todas as classes configuradas. Está aqui a saída do comando:

```
Router#show policy-map int pos6/0 POS6/0 Service-policy output: policy (1071) Class-map:
class1 (match-all) (1072/3) 0 packets, 0 bytes 5 minute offered rate 0 bps, drop rate 0 bps
Match: ip precedence 1 (1073) Class of service queue: 1 Tx Queue (DRR configured) bandwidth
percent Weight 10 1 Tx Random-detect: Exp-weight-constant: 1 (1/2) Precedence RED Label Min Max
Mark 1 1 375 2423 1 Class-map: class2 (match-all) (1076/2) 0 packets, 0 bytes 5 minute offered
rate 0 bps, drop rate 0 bps Match: ip precedence 2 (1077) Class of service queue: 2 Tx Queue
(DRR configured) bandwidth percent Weight 20 9 Tx Random-detect: Exp-weight-constant: 1 (1/2)
Precedence RED Label Min Max Mark Class-map: class-default (match-any) (1080/0) 0 packets, 0
bytes 5 minute offered rate 0 bps, drop rate 0 bps Match: any (1081) 0 packets, 0 bytes 5 minute
rate 0 bps
```

Comandos monitorar o Tratamento de Congestionamento e a vacância

Esta seção alista os comandos que você pode se usar para monitorar seus Tratamento de Congestionamento e política de fuga.

[A tabela 8](#) alista os comandos relevant para as placas de linha do ingresso e da saída.

Tabela 8 – Comandos para as placas de linha

Placa de linha do ingresso	Placa de Saída
<ul style="list-style-type: none">• show interfaces• fila do tofab do sh controller do <x> do entalhe do executivo• exec slot <x> show controller tofab queue <slot> <port>• exec slot <x> show controller tofab qm stat	<ul style="list-style-type: none">• show interfaces• mostre o <y> das relações aleatório• fila do frfab do controlador da mostra do <y> do entalhe do executivo• exec slot <y> show controller frfab queue <port>• exec slot <y> show controller frfab QM stat

Estes comandos são explicados nesta seção.

O comando show interfaces

Antes que você use este comando, confirme a “estratégia de fila correta.” Se a saída indica primeiramente dentro, primeiramente para fora (FIFO), assegure-se de que o **comando service-policy** apareça na configuração running (se o MQC foi usado para configurar o MDRR).

Monitore o número de quedas de saída, que representa o número total de quedas FrFab da WRED ocorridas para o tráfego de saída nessa interface. O número de quedas de emissor na saída do **comando show interfaces** deve ser igual a ou mais altamente do que o número de quedas de emissor na saída do **comando show interfaces <number> random**.

Nota: No Cisco 12000 Series Router, as quedas da saída de interface são atualizadas depois que as quedas do WRED forem atualizadas. Há uma possibilidade pequena que se você usa uma ferramenta para perguntar ambos os contadores de queda, as gotas da relação não estão atualizadas ainda.

```
Router#show interfaces POS 4/0 POS4/0 is up, line protocol is up Hardware is Packet over SONET
Description: link to c12f9-1 Internet address is 10.10.105.53/30 MTU 4470 bytes, BW 622000 Kbit,
DLY 100 usec, rely 255/255, load 82/255 Encapsulation PPP, crc 32, loopback not set Keepalive
set (10 sec) Scramble enabled LCP Open Open: IPCP, CDPCP, OSICP, TAGCP Last input 00:00:02,
output 00:00:05, output hang never Last clearing of "show interface" counters 00:04:54 Queueing
strategy: random early detection (WRED) Output queue 0/40, 38753019 drops; input queue 0/75, 0
drops 5 minute input rate 0 bits/sec, 0 packets/sec 5 minute output rate 200656000 bits/sec,
16661 packets/sec 135 packets input, 6136 bytes, 0 no buffer Received 0 broadcasts, 0 runts, 0
giants, 0 throttles 0 parity 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
7435402 packets output, 11182627523 bytes, 0 underruns 0 output errors, 0 applique, 0 interface
```

resets 0 output buffer failures, 0 output buffers swapped out 0 carrier transitions

[A mostra conecta o comando random do {number}](#)

Quando você usa este comando, você deve:

- Verifique que o molde correto do **cos-queue-group** está aplicado a esta relação.
- Verifique os pesos MDRR. Para cada fila MDRR, você pode verificar a média pesada para ver se há o comprimento da fila e o valor o mais alto alcançado (em uns pacotes). Os valores são calculados como uma média pesada, e não precisam de refletir o Maximum Queue Depth real alcançado nunca.
- Verifique os limiares mínimo e máximo de WRED.
- Verifique o número de perdas aleatórias e perdas de limiar para cada rótulo RED (perdas "To Fabric" indicam a quantidade total de perdas para esse rótulo em todas as placas de ingresso).
- O "TX-fila-limite deixa cair" contrários é usado somente no motor 1 LC, que não apoiam o WRED. Os cartões do motor 1 permitem-no de ajustar o limite das filas MDRR com o **comando TX-queue-limit interface**. Quando há suporte para WRED, os limiares de WRED determinam a profundidade das filas MDRR.

```
Router#show interfaces POS 4/0 random POS4/0 cos-queue-group: oc12 RED Drop Counts TX Link To
Fabric RED Label Random Threshold Random Threshold 0 29065142 73492 9614385 0 1 0 0 0 0 2 0 0 0
0 3 0 0 0 0 4 0 0 0 0 5 0 0 0 0 6 0 0 0 0 TX-queue-limit drops: 0 Queue Lengths TX Queue (DRR
configured) oc12 Queue Average High Water Mark Weight 0 0.000 2278.843 1 1 0.000 0.000 73 2
0.000 0.000 10 3 0.000 0.000 10 4 0.000 0.000 10 5 0.000 0.000 10 6 0.000 0.000 10 Low latency
0.000 0.000 10 TX RED config Precedence 0: 375 min threshold, 2423 max threshold, 1/1 mark
weight Precedence 1: not configured for drop Precedence 2: not configured for drop Precedence 3:
not configured for drop Precedence 4: 375 min threshold, 2423 max threshold, 1/1 mark weight
Precedence 5: not configured for drop Precedence 6: 375 min threshold, 2423 max threshold, 1/1
mark weight Precedence 7: not configured for drop weight 1/2
```

[O entalhe do executivo \(y\) comando do {port} da fila do frfab do controlador da mostra](#)

Este comando indica a profundidade de fila instantânea para uma porta dada em um entalhe dado. O exemplo de saída nesta seção indica a fila MDRR na relação POS 4/1. Você vê uma profundidade de fila para a fila 1 MDRR de 1964 pacotes. O peso é o número de bytes que pode ser servido nesta fila. Este peso determina o porcentagem de largura de banda que você quer dar a esta fila. O deficit é o valor que diz ao algoritmo DRR quantos pacotes ainda precisam de ser servidos. Você pode ver que não há nenhum pacote enfileirado no LLQ (fila 7 DRR).

```
Router#execute-on slot 4 show controllers frfab queue 1 ===== Line Card (Slot 4) =====
FrFab Queue Interface 1 DRR# Head Tail Length Average Weight Deficit 0 95330 40924 0 0.000 4608
0 1 211447 233337 1964 1940.156 41472 35036 2 0 0 0 0.000 9216 0 3 0 0 0 0.000 9216 0 4 0 0 0
0.000 9216 0 5 0 0 0 0.000 9216 0 6 0 0 0 0.000 9216 0 7 0 0 0 0.000 9216 0
```

Esse comando é utilizado principalmente para monitorar a profundidade da Fila de prioridade da placa de linha de saída. Quando você vê que os pacotes começam esperar neste LLQ, é uma boa indicação que você deve desviar alguma Voz sobre o tráfego IP (VOIP) a outras placas de linha da saída. Em um bom projeto, o comprimento deve sempre ser 0 ou 1. Em uma rede de vida real, você experimentará o tráfego intermitente, mesmo para dados de voz. O retardo extra se torna mais grave quando a carga de voz total excede 100% da largura de banda de entrada para um intervalo curto. O roteador não pode colocar mais tráfego na rede do que o permitido, por isso, o tráfego de voz é enfileirado em sua própria fila de prioridade. Isto cria a latência da Voz e o tremor da Voz introduzidos pela explosão do tráfego de voz própria.

```
Router#execute-on slot 4 show controllers frfab queue 0 ===== Line Card (Slot 4) =====
FrFab Queue Interface 0 DRR# Head Tail Length Average Weight Deficit 0 181008 53494 2487
2282.937 4608 249 1 16887 45447 7 0.000 41472 0 2 0 0 0 0.000 9216 0 3 0 0 0 0.000 9216 0 4 0 0
0 0.000 9216 0 5 0 0 0 0.000 9216 0 6 0 0 0 0.000 9216 0 7 107818 142207 93 0.000 9216 -183600
```

A fila 7 é o LLQ, e o comprimento diz-lhe quantos pacotes estão neste LLQ.

[O entalhe do executivo \(y\) comando qm stat do frfab do controlador da mostra](#)

Use este comando quando você suspeita que a memória de pacotes de um LC começa a aproximar a capacidade completa. Um valor crescente para “nenhuma gota do mem” contrária sugere que o WRED não esteja configurado ou que os limites de WRED estão ajustados demasiado altos. Este contador não deve incrementar em condições normais. Não veja [pacotes ignorados do Troubleshooting e nenhuma gota da memória no Cisco 12000 Series Internet Router](#) para mais informação.

```
Router#execute-on slot 4 show controllers frfab QM stat ===== Line Card (Slot 4) =====
68142538 no mem drop, 0 soft drop, 0 bump count 0 rawq drops, 8314999254 global red drops,
515761905 global force drops 0 no memory (ns), 0 no memory hwm (Ns) no free queue 0 0 1968 88 0
0 0 0 0 0 0 0 0 0 0 0 0 0 multicast drops TX Counts Interface 0 859672328848 TX bytes, 3908130535
TX pkts, 75431 kbps, 6269 pps Interface 1 86967615809 TX bytes, 57881504 TX pkts, 104480 kbps,
8683 PPS Interface 2 0 TX bytes, 0 TX pkts, 0 kbps, 0 PPS Interface 3 0 TX bytes, 0 TX pkts, 0
kbps, 0 PPS
```

[Monitore o Tratamento de Congestionamento de entrada](#)

Esta seção descreve os comandos usados para monitorar o Tratamento de Congestionamento de entrada.

[O comando show interfaces](#)

Antes que você emita este comando, verifique se o valor no contador ignorado esteja no aumento. Você verá pacotes ignorados se você executado fora da memória no lado do tofab ou se a placa de linha não aceita os pacotes rapidamente bastante. [Para obter mais informações, consulte Troubleshooting de Quedas de Entrada no Cisco 12000 Series Router.](#)

```
Router#show interfaces POS 14/0 POS14/0 is up, line protocol is up Hardware is Packet over SONET
Description: agilent 3b for QOS tests Internet address is 10.10.105.138/30 MTU 4470 bytes, BW
2488000 Kbit, DLY 100 usec, rely 234/255, load 1/255 Encapsulation HDLC, crc 32, loopback not
set Keepalive not set Scramble disabled Last input never, output 00:00:03, output hang never
Last clearing of "show interface" counters 00:34:09 Queueing strategy: random early detection
(WRED) Output queue 0/40, 0 drops; input queue 0/75, 0 drops 5 minute input rate 2231000
bits/sec, 4149 packets/sec 5 minute output rate 0 bits/sec, 0 packets/sec 563509152 packets
input, 38318622336 bytes, 0 no buffer Received 0 broadcasts, 0 runts, 0 giants, 0 throttles 0
parity 166568973 input errors, 0 CRC, 0 frame, 0 overrun, 166568973 ignored, 0 abort 35 packets
output, 12460 bytes, 0 underruns 0 output errors, 0 applique, 0 interface resets 0 output buffer
failures, 0 output buffers swapped out 0 carrier transitions
```

[O comando show controller tofab queue do slot\(x\) do executivo](#)

Este exemplo de saída do comando `exec slot <x> show controller tofab queue` foi capturado quando não havia nenhuma congestão em uma placa de linha da saída no entalhe 3.

```
Router#execute-on slot 13 show controllers tofab queue ===== Line Card (Slot 13) =====
Carve information for ToFab buffers !--- Output omitted. ToFab Queues: Dest Slot 0 0 0 0 9690 1
0 0 0 9690 2 0 0 0 9690 3 11419 16812 0 9690 4 0 0 0 2423 5 0 0 0 9690 6 0 0 0 9690 7 0 0 0
262143 8 0 0 0 262143 9 0 0 0 606 10 0 0 0 262143 11 0 0 0 262143 12 0 0 0 262143 13 0 0 0
262143 14 0 0 0 262143 15 0 0 0 9690 Multicast 0 0 0 262143
```

A seguinte saída foi capturada quando havia uma congestão no entalhe 3:

```
Router#execute-on slot 13 show controllers tofab queue ===== Line Card (Slot 13) =====
Carve information for ToFab buffers !--- Output omitted. ToFab Queues: Dest Slot 0 0 0 0 9690 1
0 0 0 9690 2 0 0 0 9690 3 123689 14003 1842 9690 4 0 0 0 2423 5 0 0 0 9690 6 0 0 0 9690 7 0 0 0
262143 8 0 0 0 262143 9 0 0 0 606 10 0 0 0 262143 11 0 0 0 262143 12 0 0 0 262143 13 0 0 0
262143 14 0 0 0 262143 15 0 0 0 9690 Multicast 0 0 0 262143
```

O comando da fila do tofab do controlador da mostra do slot(x) do executivo (entalhe) (porta)

Use este comando determinar quanto memória é usada no lado do tofab. Em particular, note o número coluna 'do #Qelem na". Observe isso:

- Quando nenhuma memória é usada, os valores estão no seu mais alto.
- O valor da coluna do "#Qelem" diminui enquanto os pacotes são protegidos.
- Quando a coluna "#Qelem" atingir zero, todos os buffers gravados estarão em uso. No Engine 2 LC, pequenos pacotes podem emprestar espaço de buffer de pacotes maiores.

Você pode igualmente usar este comando determinar o número de pacotes em fila em uma fila de saída virtual. O exemplo aqui mostra como verificar o entalhe 14 para ver se há o número instantâneo de pacotes nestas filas para o entalhe 4, a porta 1 (POS 4/1). Nós vemos 830 pacotes enfileirados na fila 1. MDRR.

```
Router# execute-on slot 14 show controllers tofab queue 4 1 ===== Line Card (Slot 14)
===== ToFab Queue Slot 4 Int 1 DRR# Head Tail Length Average Weight Deficit 0 0 0 0 0.000 4608
0 1 203005 234676 830 781.093 41472 37248 2 0 0 0 0.000 9216 0 3 0 0 0 0.000 9216 0 4 0 0 0
0.000 9216 0 5 0 0 0 0.000 9216 0 6 0 0 0 0.000 9216 0 7 0 0 0 0.000 9216 0
```

O comando qm stat do tofab do controlador da mostra do slot(x) do executivo

Use esse comando para ver o número de quedas ToFab por placa de ingresso. Igualmente verifique para ver se há de "um esse contrário nenhuma gota da memória" incrementos. Esse contador é incrementado quando o CoS não está configurado no ToFab.

```
Router#execute-on slot 13 show controllers tofab QM stat ===== Line Card (Slot 13) =====
no mem drop, 0 soft drop, 0 bump count 0 rawq drops, 1956216536 global red drops, 6804252 global
force drops 0 no memory (Ns), 0 no memory hwm (Ns) no free queue 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Q status errors 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Casos Práticos

Estes Casos Práticos mostram como configurar uma política típica para o centro de rede de um ambiente de provedor de serviços. Aplicam comandos queue e permitem-no de usar o MDRR/WRED para o gerenciamento de fila ativo. As políticas de QoS nos roteadores de ponta usam normalmente a marcação do tráfego, condicionando, e assim por diante, para permitir o Roteadores no núcleo de classificar o tráfego nas classes baseadas em valores da Precedência IP ou do DiffServ Code Point (DSCP). Estes Casos Práticos usam características de QoS do Cisco IOS Software para encontrar o Service Level Agreements apertado (SLA) e níveis de serviço diferentes para a Voz, o vídeo, e os serviços dos dados no mesmo backbone IP.

Na aproximação, um provedor de serviços executou três classes de tráfego. A mais importante é a LLQ ou a classe de enfileiramento de latência baixa. Essa é a classe para Voz e Vídeo. Esta classe deve experimentar um retardo mínimo e um tremor, e deve nunca experimentar a perda de pacotes ou pacotes requisitados novamente enquanto a largura de banda desta classe não excede a largura de banda de enlace. Essa classe é conhecida como tráfego EF PHB (Expedited

Forwarding Per-Hop Behavior) na arquitetura DiffServ. O provedor de serviço do Internet (ISP) projetou a rede em uma maneira que esta classe não excedesse 30% na carga média da largura de banda de enlace. As outras duas classes são a classe de negócios e a classe de melhor esforço.

No projeto, nós configuramos o Roteadores de tal maneira que a classe de negócio obtém sempre 90% da largura de banda remanescente e a classe de melhor esforço obtém 10%. Estas duas classes têm menos tráfego sensível ao tempo e podem experimentar a perda de tráfego, um atraso mais alto, e um tremor. No projeto, o foco está em placas de linha do Engine 2: 1xOC48 rev B, 4xOC12 rev as placas de linha B, e 8xOC3.

As placas de linha do Rev B são seridas melhor para levar o tráfego voip devido a um ASIC e a uma arquitetura de hardware revisados, que introduza a latência muito pequena. Com o ASIC revisado, a fila de FIFO transmitir resized pelo direcionador da placa de linha a aproximadamente duas vezes o MTU o maior no cartão. Procure “- B” adicionou ao part number, tal como o OC48E/POS-SR-SC-B=.

Nota: Não confunda a fila FIFO de transmissão com as filas FrFab que podem ser sintonizadas nas placas de ingresso do Mecanismo 0 com o comando tx-queue-limit interface.

[Apresente 9](#) alista os critérios correspondentes para cada classe.

Tabela 9 – Critérios correspondentes para cada classe

Nome de classe	Critérios de correspondência
Fila de Prioridade - Tráfego de voz	Precedência 5
Fila de negócios	Procedência 4
A melhor fila do esforço	Precedência 0

As placas de linha OC48 podem enfileirar um grande número de pacotes nas filas ToFab. Assim, é importante configurar o MDRR/WRED nas filas ToFab, especialmente quando a interface de saída é uma interface de alta velocidade, como a OC48. A tela pode comutar apenas o tráfego para a placa de linha de recepção, em uma taxa máxima teórica de 3 Gbps (pacotes de 1500 bytes). Se a quantidade total de tráfego enviada é maior do que o que a tela de switching pode levar a seu cartão de recepção, muitos pacotes serão enfileirados nas filas do tofab.

Interface POS3/0

```

description OC48 egress interface ip address 10.10.105.53 255.255.255.252 no ip directed-
broadcast ip router Isis encapsulation ppp mpls traffic-eng tunnels tag-switching ip no peer
neighbor-route crc 32 clock source internal POS framing sdh POS scramble-atm POS threshold sf-
ber 4 POS flag s1s0 2 TX-cos oc48 Isis metric 2 level-1 Isis metric 2 level-2 ip rsvp bandwidth
2400000 2400000 ! interface POS4/1 description OC12 egress interface ip address 10.10.105.121
255.255.255.252 no ip directed-broadcast ip router Isis encapsulation ppp mpls traffic-eng
tunnels no peer neighbor-route crc 32 clock source internal POS framing sdh POS scramble-ATM POS
threshold sf-ber 4 POS flag s1s0 2 TX-cos oc12 Isis metric 2 level-1 Isis metric 2 level-2 ip
RSVP bandwidth 600000 60000 ! interface POS9/2 description OC3 egress interface ip address
10.10.105.57 255.255.255.252 no ip directed-broadcast ip router Isis crc 16 POS framing sdh POS
scramble-ATM POS flag s1s0 2 TX-cos oc3 Isis metric 200 level-1 Isis metric 2 level-2 !
interface POS13/0 description agilent 3a for QOS tests - ingress interface. ip address
10.10.105.130 255.255.255.252 no ip directed-broadcast no ip route-cache cef no ip route-cache
no ip mroute-cache no keepalive crc 32 POS threshold sf-ber 4 TX-cos oc48 ! interface POS14/0
description agilent 3b for QOS tests - ingress interface. ip address 10.10.105.138
255.255.255.252 no ip directed-broadcast no keepalive crc 32 POS threshold sf-ber 4 TX-cos oc48
! interface POS15/0 description agilent 4A for QOS tests - ingress interface ip address

```



```
10.10.105.134 255.255.255.252 no ip directed-broadcast no ip mroute-cache no keepalive crc 32
POS threshold sf-ber 4 TX-CoS oc48 ! rx-cos-slot 3 StotTable rx-cos-slot 4 StotTable rx-cos-slot
9 StotTable rx-cos-slot 13 StotTable rx-cos-slot 14 StotTable rx-cos-slot 15 StotTable ! slot-
table-cos StotTable destination-slot 0 oc48 destination-slot 1 oc48 destination-slot 2 oc48
destination-slot 3 oc48 destination-slot 4 oc12 destination-slot 5 oc48 destination-slot 6 oc48
destination-slot 9 oc3 destination-slot 15 oc48 ! cos-queue-groupoc3 precedence 0 random-detect-
label 0 precedence 4 queue 1 precedence 4 random-detect-label 1 precedence 5 queue low-latency
precedence 6 queue 1 precedence 6 random-detect-label 1 random-detect-label 0 94 606 1 random-
detect-label 1 94 606 1 queue 0 1 queue 1 73 queue low-latency strict-priority !--- Respect the
tight SLA requirements. !--- No packets drop/low delay and jitter for the priority queue. ! CoS-
queue-groupoc12 precedence 0 random-detect-label 0 precedence 4 queue 1 precedence 4 random-
detect-label 1 precedence 5 queue low-latency precedence 6 queue 1 precedence 6 random-detect-
label 1 random-detect-label 0 375 2423 1 random-detect-label 1 375 2423 1 queue 0 1 queue 1 73
queue low-latency strict-priority ! CoS-queue-groupoc48 precedence 0 random-detect-label 0
precedence 4 queue 1 precedence 4 random-detect-label 1 precedence 5 queue low-latency
precedence 6 queue 1 precedence 6 random-detect-label 1 random-detect-label 0 1498 9690 1
random-detect-label 1 1498 9690 1 queue 0 1 queue 1 73 queue low-latency strict-priority
```

Espera-se que mais tráfego voip que você tem, mais o tráfego do negócio tenha que esperar antes que obtém servido. Contudo, este não é um problema porque o SLA apertado não exige nenhuma queda de pacote de informação, e muito latência baixa e tremor para a fila de prioridade.

[Informações Relacionadas](#)

- [Como ler a saída do frfab do controlador da mostra | comandos de fila tofab em um Cisco 12000 Series Internet Router](#)
- [Troubleshooting de Pacotes Ignorados e Quedas Sem Memória no Cisco 12000 Series Internet Router](#)
- [Troubleshooting de Quedas de Entrada no Cisco 12000 Series Internet Router](#)
- [Detecção antecipada aleatória ponderada no Cisco 12000 Series Router](#)
- [Vista geral da interface da linha de comando da Qualidade de Serviço modular](#)
- [Página de suporte dos 12000 Series Internet Router](#)
- [Suporte Técnico - Cisco Systems](#)