Compare a política de tráfego e a forma do tráfego para limitar a largura de banda

Contents

Introdução

Este documento descreve as diferenças funcionais entre modelagem de tráfego e vigilância de tráfego, que limitam a taxa de saída.

Pré-requisitos

Requisitos

Não existem requisitos específicos para este documento.

Componentes Utilizados

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

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 rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Conventions

Consulte as Convenções de Dicas Técnicas da Cisco para obter mais informações sobre

convenções de documentos.

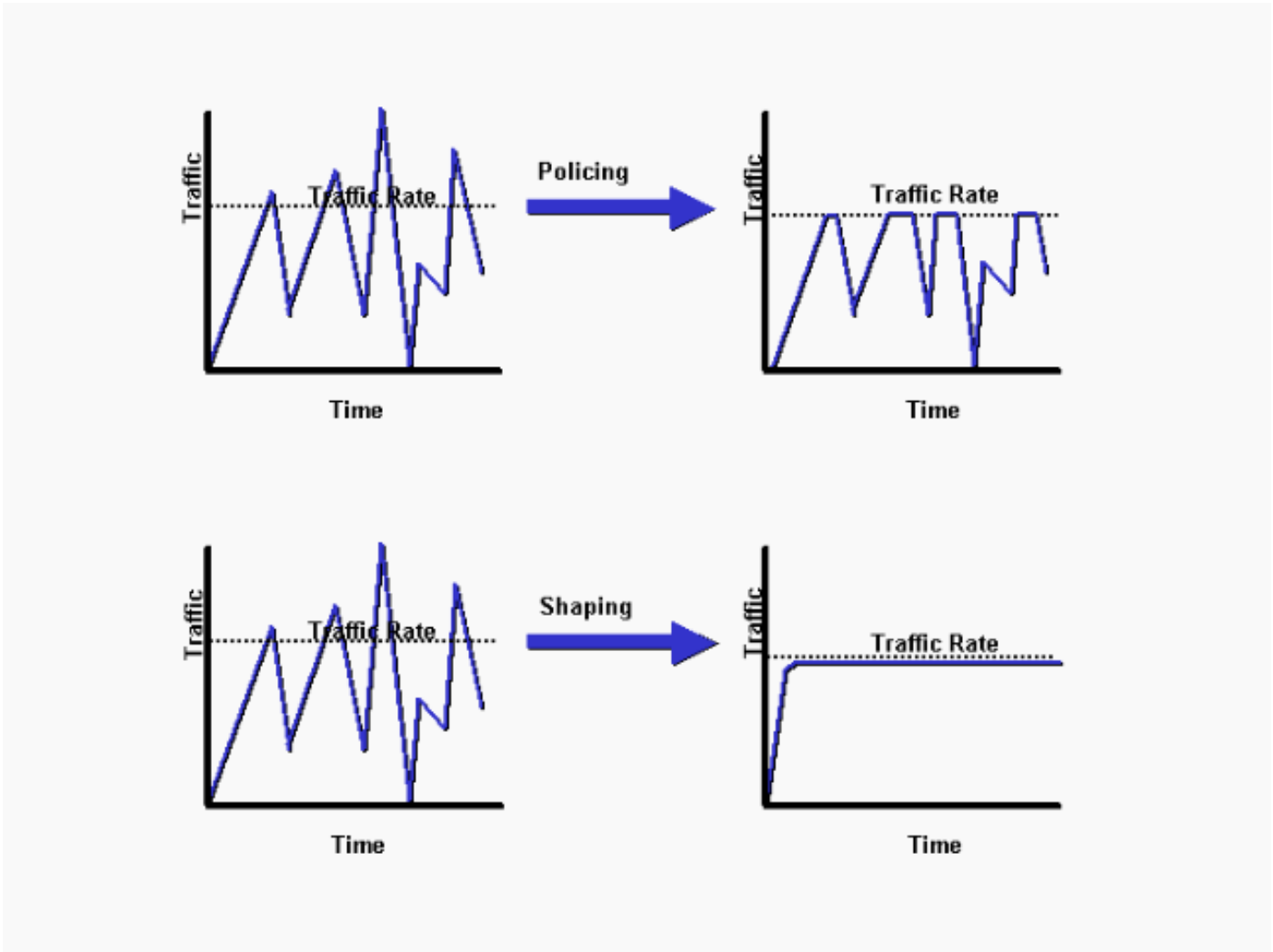
Informações de Apoio

Este documento esclarece as diferenças funcionais entre modelagem de tráfego e vigilância. Ambas limitam funcionalmente a taxa de saída de tráfego. Ambos os mecanismos usam um token bucket como medidor de tráfego para medir a taxa de pacotes. Para obter mais informações sobre token buckets, consulte O que é um token bucket

Vigilância versus modelagem

A política de tráfego propaga os bursts. Quando a taxa de tráfego atinge a taxa máxima configurada, o tráfego de excesso é liberado (ou remarcado). O resultado é uma taxa de saída que aparece como um dente de serra com picos e depressões. Em contraste com a vigilância, a modelagem de tráfego retém pacotes em excesso em uma fila e agenda o excesso para transmissão posterior de acordo com incrementos de tempo. O resultado da modelagem de tráfego é uma taxa de saída de pacote facilitada.

O próximo diagrama ilustra as principais diferenças entre as duas opções de tráfego.



Vigilância VS Modelagem

A modelagem implica na existência de uma fila e de memória suficiente para armazenar os pacotes com retardo em buffer, diferente da vigilância. As filas são um conceito de saída; os pacotes que saem de uma interface são enfileirados e podem ser moldados. Somente a política pode ser aplicada ao tráfego de entrada em uma interface. Certifique-se de ter memória suficiente ao habilitar a modelagem. Além disso, a modelagem requer uma função que programe a transmissão posterior de qualquer pacote atrasado. Essa funcionalidade de agendamento permite organizar a fila de modelagem em filas diferentes. Exemplos dessa funcionalidade são Class Based Weighted Fair Queuing (CBWFQ) e Low Latency Queuing (LLQ).

Critérios de seleção

A tabela a seguir lista as diferenças entre modelagem e vigilância para ajudá-lo a escolher a solução de tráfego apropriada.

	Modelagem	Vigilância
Objetivo	Armazene em buffer e enfileire pacotes em excesso nas taxas comprometidas.	Descarte (ou remarque) pacotes em excesso sobre as taxas comprometidas. Não é possível armazenar em buffer.*
Taxa de atualização de token	Incrementada no início de um intervalo de tempo. (É necessário um número mínimo de intervalos.)	Contínuo com base na fórmula: `1 / taxa de informação comprometida`
Valores de token	Configurado em bits por segundo.	Configurado em bytes.
Opções de configuração	• shape comando na Interface de linha de comando de qualidade de serviço modular (MQC, Modular Quality of Service Command-Line Interface) para implementar a class-based shaping. • Comando frame-relay traffic-shape para implementar modelagem de tráfego de Frame Relay (FRTS). • comando traffic-shape para implementar o Generic Traffic Shaping (GTS).	• Comando police no MQC para implementar a vigilância baseada em classe. • comando rate-limit para implementar a Taxa de acesso consolidada (CAR).
Aplicável na Entrada	No	Yes
Aplicável na Saída	Yes	Yes
Intermitências	Controla rajadas e suaviza a taxa de saída em pelo menos oito intervalos de tempo. Utiliza um vazamento de bucket para retardar tráfego, o que causa um	Propaga intermitências. Não faz suavização.

	efeito facilitador.	
Vantagens	Menor probabilidade de descartar pacotes em excesso, visto que estes sofrem buffer. (Pacotes de buffer até o comprimento da fila. Poderão ocorrer quedas se o excesso de tráfego for mantido em altas taxas.) Normalmente evita retransmissões devido a pacotes descartados.	Controla a taxa de saída por meio do cancelamento de pacotes. Evita atrasos devidos a queuing.
Desvantagens	Pode introduzir atraso devido a filas queuingparticularmente profundas.	Descarta pacotes em excesso (quando configurados), tamanhos de janela TCP acelerados e reduz a taxa de saída geral dos fluxos de tráfego afetados. Tamanhos de burst excessivamente agressivos podem levar a quedas de pacotes em excesso e acelerar a taxa de saída geral, particularmente com fluxos baseados em TCP.
Remarcação de Pacote Opcional	No	Sim (com recurso de CAR legado).

* Embora a vigilância não aplique um buffer, um queuing mecanismo configurado se aplica a pacotes conformes que precisam ser enfileirados enquanto esperam para serem serializados na interface física.

Taxa de atualização de token

Uma diferença importante entre modelagem e política é a taxa à qual os tokens são recarregados. A modelagem e a vigilância usam a metáfora do token bucket. Um token bucket propriamente dito não possui política de descarte ou prioridade.

Com funcionalidade de token bucket:

- Os tokens são colocados em um bucket a uma certa taxa.
- Cada token dá permissão para que a origem envie um determinado número de bits para a rede.
- Para enviar um pacote, o regulador de tráfego deve estar apto a remover do bucket um número de símbolos que seja igual em representação ao tamanho do bucket.

- Se não houver tokens suficientes no bucket para enviar um pacote, o pacote esperará até que o bucket tenha tokens suficientes (no caso de um modelador) ou o pacote será descartado ou marcado (no caso de um vigilante).

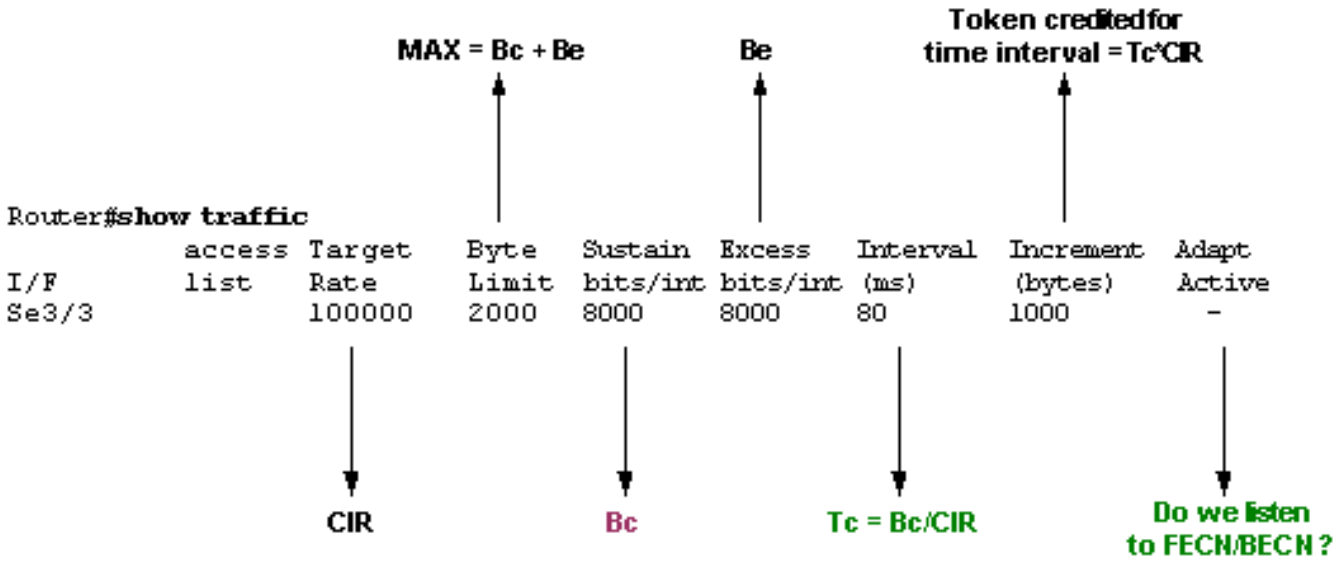
- O bucket propriamente dito possui uma capacidade específica. Se o bucket for preenchido até a capacidade máxima, os novos tokens que chegarem serão descartados e não estarão disponíveis para pacotes futuros. Assim, a qualquer momento, o maior surto que uma origem pode enviar na rede é proporcional ao tamanho do pacote. Um token bucket permite intermitência, mas a limita.

A modelagem incrementa o token bucket em intervalos programados que usam um valor de bits por segundo (bps). Um modelador usa a próxima fórmula:

```
Tc = Bc/CIR (in seconds)
```

Nessa equação, Bc representa o burst comprometido, e CIR quer dizer Taxa de informação comprometida (Committed Information Rate). (Consulte a Configuração da formatação de tráfego frame relay para obter mais informações.) O valor de Tc define o intervalo de tempo durante o qual você envia os bits Bc para manter a taxa média da CIR em segundos.

O intervalo de Tc é entre 10 ms e 125 ms. Com o Distributed Traffic Shaping (DTS) no Cisco 7500 Series, o Tc mínimo é de 4 ms. O roteador calcula internamente esse valor com base nos valores da CIR e do Bc. Se Bc/CIR for menor que 125 ms, ele utiliza o Tc calculado a partir daquela equação. Se o Bc/CIR for maior ou igual a 125 ms, ele usará um valor de Tc interno se o Cisco IOS® determinar que o fluxo de tráfego pode ser mais estável com um intervalo menor. Use o comando **show traffic-shape** para determinar se o roteador usa um valor interno para Tc ou o valor que você configurou na linha de comando. O próximo exemplo de saída do comando **show traffic-shape** é explicado em Comandos show para modelagem de tráfego de Frame Relay.



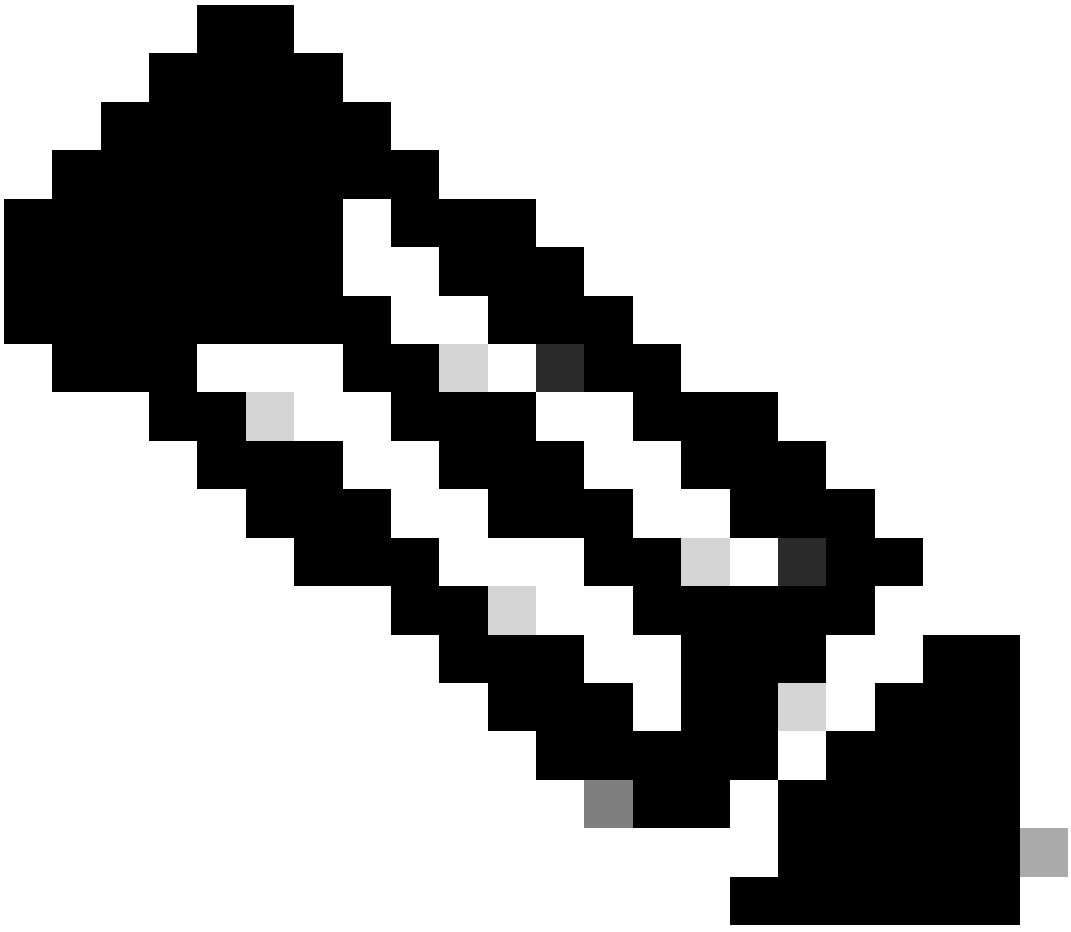
show traffic output

Quando o excesso de burst (Be, excess burst) é configurado para um valor diferente de 0, o modelador permite que os tokens sejam armazenados no bucket até Bc + Be. O maior valor que o token bucket pode alcançar é Bc + Be, e os tokens de sobrefluxo são descartados. A única maneira de ter mais tokens Bc no bucket é não utilizar todos os tokens Bc durante um ou mais Tcs. Como o token bucket é repovoado a cada Tc com tokens Bc, é possível acumular tokens não utilizados para uso posterior em Bc + Be.

Em contraste, a vigilância baseada em classe e a taxa limiting adicionam tokens continuamente ao bucket. Especificamente, a taxa de chegada do token é calculada da seguinte maneira:

```
(time between packets<which is equal to t-t1>* policer rate)/8 bits per byte
```

Em outras palavras, se a chegada anterior do pacote é t1 e o horário atual é t, o bucket é atualizado com o valor de bytes de t-t1 com base na taxa de chegada do token.



Observação: um vigilante de tráfego usa valores de intermitência especificados em bytes e a fórmula anterior converte de bits em bytes.

Este é um exemplo que usa uma CIR (ou taxa de vigilante) de 8000 bps e uma intermitência normal de 1000 bytes:

```
<#root>

Router(config)#

policy-map police-setting

Router(config-pmap)#

class access-match

Router(config-pmap-c)#

police 8000 1000 conform-action transmit exceed-action drop
```

Os token buckets começam cheios em 1000 bytes. Se um pacote com 450 bytes chegar, ele é aceito, pois há bytes suficientes disponíveis no token bucket. A ação de conformidade (transmissão) é realizada pelo pacote e 450 bytes são removidos do token bucket (e deixam 550 bytes). Se o próximo pacote chegar 0,25 segundo depois, 250 bytes serão adicionados ao token bucket de acordo com a próxima fórmula:

```
(0.25 * 8000)/8
```

O cálculo deixa 700 bytes no token bucket. Se o próximo pacote tiver 800 bytes, o pacote implicará excesso e a ação exceder (cancelamento) será executada. Nenhum byte foi retirado do token bucket.

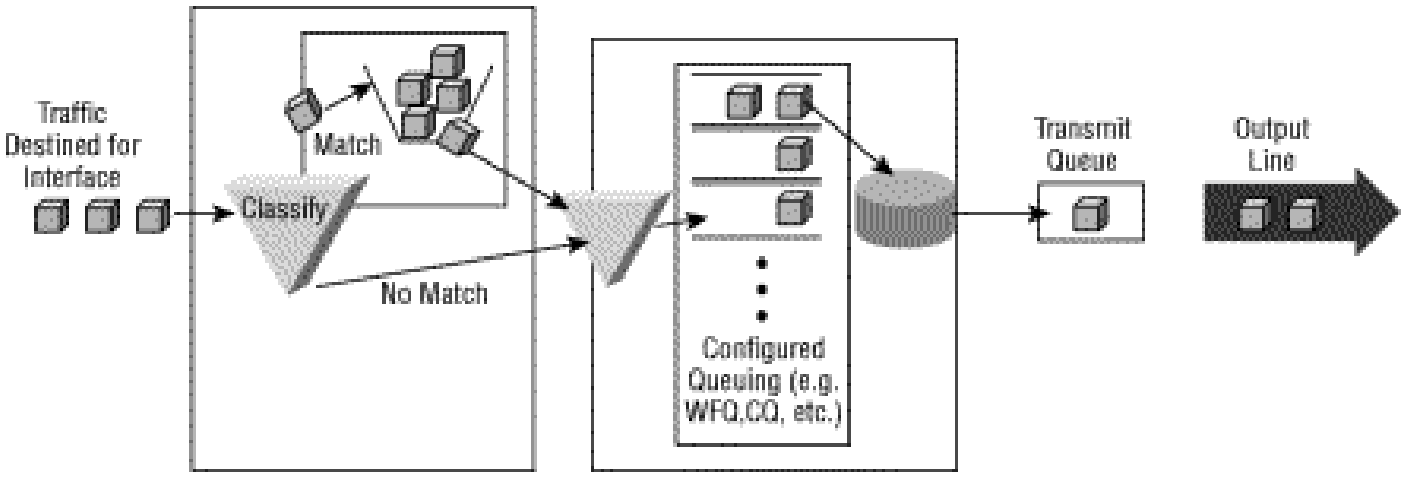
Modelagem de tráfego

O Cisco® IOS suporta os próximos métodos de modelagem de tráfego:

- Formatação de tráfego genérico
- Formatação de tráfego frame relay
- Modelagem com base em classes e Modelagem Distribuída com base em classes

Todos os métodos de modelagem são semelhantes em implementação, embora suas interfaces de linha de comandos (CLIs) sejam um pouco diferentes e usem tipos diferentes de filas para conter e modelar o tráfego adiado. A Cisco recomenda modelagem baseada em classe e modelagem distribuída, que são configuradas com a CLI QoS modular.

O próximo diagrama ilustra como uma política de QoS organiza o tráfego em classes e enfileira pacotes que excedem as taxas de modelagem configuradas.



Políticas de tráfego

O Cisco IOS suporta os próximos métodos de vigilância de tráfego:

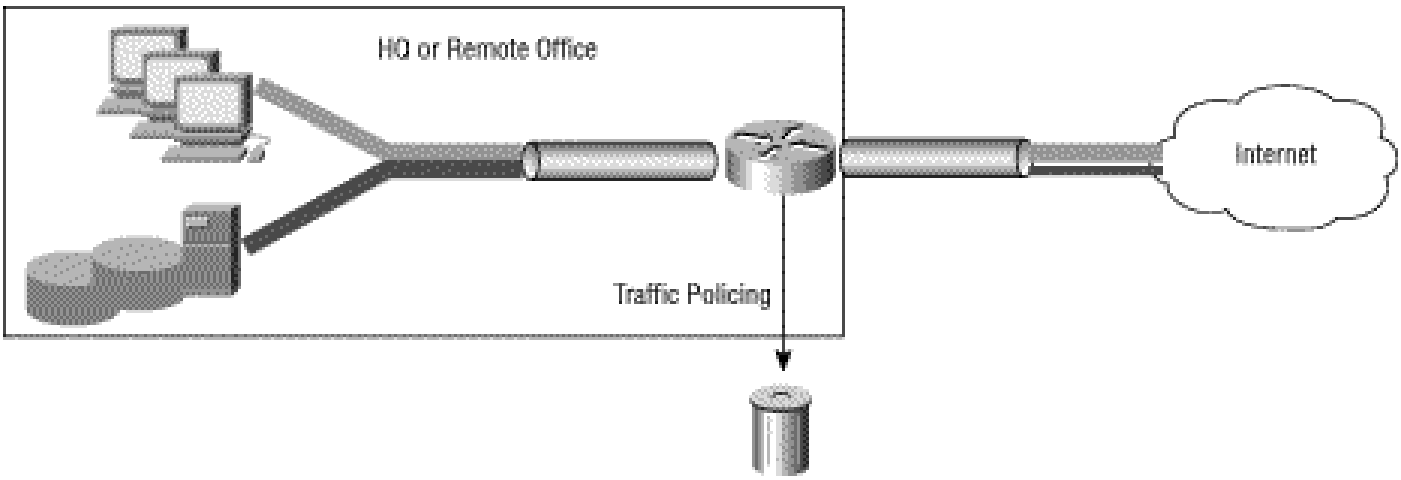
- Taxa de acesso consolidada
- Vigilância baseada em classe

Os dois mecanismos têm diferenças funcionais importantes, conforme explicado em Compare Class Based Policing e Committed Access Rate. A Cisco recomenda a vigilância baseada em classe e outros recursos da CLI QoS modular quando as políticas de QoS são aplicadas.

Use o comando **police** para especificar que uma classe de tráfego deve ter uma taxa máxima imposta a ela e, se essa taxa for excedida, uma ação imediata deverá ser tomada. Ou seja, com o comando de vigia, não é possível armazenar o pacote no buffer e, posteriormente, enviá-lo, como é o caso do comando de modelagem.

Além disso, com a vigilância, o token bucket determina se um pacote excede ou corresponde à taxa aplicada. Em ambos os casos, a vigilância implementa uma ação configurável, que inclui a precedência de IP ou o Ponto de Código de Serviços Diferenciados (DSCP).

O próximo diagrama ilustra uma aplicação comum de vigilância de tráfego em um ponto de congestionamento, onde os recursos de QoS geralmente se aplicam.



Controles de largura de banda mínima versus máxima

Os comandos shape e police restringem a taxa de saída a um valor máximo em kbps. O mais importante é que nenhum dos mecanismos fornece uma garantia de largura de banda mínima durante períodos de congestionamento. Use o comando bandwidth ou priority para fornecer essas garantias.

Uma política hierárquica usa duas políticas de serviço - uma política pai para aplicar um mecanismo de QoS a um agregado de tráfego e uma política filha para aplicar um mecanismo de QoS a um fluxo ou subconjunto do agregado. Interfaces lógicas, como subinterfaces e interfaces de túnel, exigem uma política hierárquica com o limiting recurso de tráfego no nível pai e enfileiramento em níveis inferiores. O limiting recurso de tráfego reduz a taxa de saída e (presumivelmente) cria congestionamento, conforme visto pelos pacotes em queuing excesso.

A próxima configuração é sub-ótima e é mostrada para ilustrar a diferença entre o comando **police** e o comando **shape** quando limiting um tráfego é agregado, neste caso, class-default, a uma taxa máxima. Nessa configuração, o comando **police** envia pacotes das classes filho com base no tamanho do pacote e no número de bytes que permanecem nos token buckets de conformidade e excedente. (Consulte Política de tráfego.) O resultado é que as taxas dadas às classes de Voz sobre IP (VoIP) e Protocolo de Internet (IP) não podem ser garantidas, já que o recurso de polícia substitui as garantias feitas pelo recurso de prioridade.

Entretanto, se o comando shape for utilizado, o resultado será um sistema de enfileiramento hierárquico, e todas as garantias serão implementadas. Em outras palavras, quando a carga oferecida excede a taxa de forma, as classes VoIP e IP têm garantia de sua taxa, e o tráfego padrão da classe (no nível infantil) incorre em qualquer queda.

Cuidado: essa configuração não é recomendada e é mostrada para ilustrar a diferença entre o comando **police** versus o comando **shape quando ele limita um agregado de tráfego.**

```
class-map match-all IP
   match ip precedence 3
class-map match-all VoIP
   match ip precedence 5

policy-map child
  class VoIP
     priority 128
   class IP
     priority 1000

 policy-map parent
   class class-default
     police 3300000 103000 103000 conform-action transmit exceed-action drop
     service-policy child
```

Para que a configuração anterior faça sentido, a vigilância deve ser substituída pela modelagem.

Por exemplo:

```
policy-map parent
    class class-default
     shape average 3300000 103000 0
      service-policy child
```

Observação: para saber mais sobre políticas pai e filho, consulte Política de Serviço Filho de QoS para Classe de Prioridade .

Informações Relacionadas

- **Suporte à tecnologia de qualidade de serviços (QoS)**
- **Suporte Técnico e Documentação - Cisco Systems**

Sobre esta tradução
A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês (link fornecido) seja sempre consultado.