

Comparación de la Regulación del Tráfico y el Modelado del Tráfico para la Limitación del Ancho de Banda

Contenido

[Introducción](#)

[Antes de comenzar](#)

[Convenciones](#)

[prerrequisitos](#)

[Componentes Utilizados](#)

[Modelado versus regulación del tráfico](#)

[Criterios de selección](#)

[Velocidad de actualización de token](#)

[Modelado de tráfico](#)

[Regulación del tráfico](#)

[Controles de ancho de banda mínimo versus máximo](#)

[Información Relacionada](#)

[Introducción](#)

Este documento clarifica las diferencias funcionales entre el modelado y la regulación, los cuales limitan la velocidad de salida. Pese a que ambos mecanismos utilizan una cubeta con ficha como medidor de tráfico para medir la velocidad de los paquetes, tienen diferencias funcionales importantes. (La cubeta con ficha se describe en [What Is a Token Bucket?](#) (¿qué es una cubeta con ficha?).

[Antes de comenzar](#)

[Convenciones](#)

Consulte [Convenciones de Consejos TécnicosCisco](#) para obtener más información sobre las convenciones del documento.

[prerrequisitos](#)

No hay requisitos previos específicos para este documento.

[Componentes Utilizados](#)

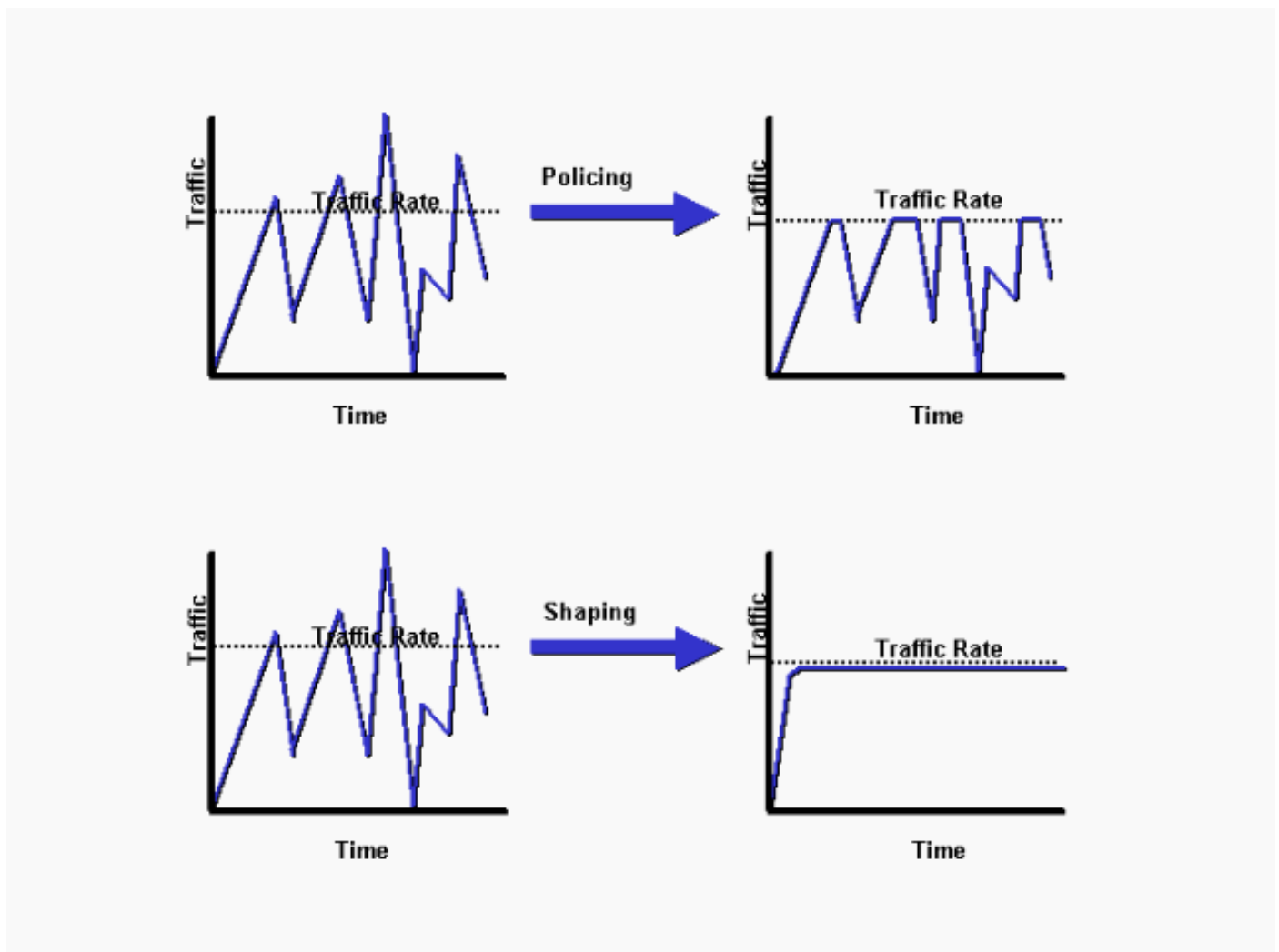
Este documento no tiene restricciones específicas en cuanto a versiones de software y de

hardware.

La información que se presenta en este documento se originó a partir de dispositivos dentro de un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si la red está funcionando, asegúrese de haber comprendido el impacto que puede tener un comando antes de ejecutarlo.

Modelado versus regulación del tráfico

El siguiente diagrama ilustra la diferencia fundamental. La regulación del tráfico propaga ráfagas. Cuando la velocidad del tráfico alcance la velocidad máxima configurada, el tráfico en exceso se suprime (o remarca). El resultado es una velocidad de salida que tiene la apariencia de un diente de sierra, con crestas y depresiones. A diferencia del establecimiento de políticas, el modelado del tráfico retiene los paquetes excedentes en una cola y luego programa dicho excedente para su posterior transmisión en incrementos de tiempo. El resultado del diseño del tráfico es una velocidad atenuada del paquete de salida.



A diferencia del establecimiento de políticas, el modelado implica la existencia de una cola y de memoria suficiente para almacenar en una memoria intermedia los paquetes con retraso. El envío a cola es un concepto de salida; los paquetes que salen de una interfaz se ponen en cola y se pueden modelar. Solamente se puede aplicar una regulación al tráfico entrante en una interfaz. Asegúrese de que tiene memoria suficiente cuando habilite el modelado. Además, el modelado requiere una función de programación para la posterior transmisión de cualquier paquete

demorado. Esta función de programación le permite organizar la cola de modelado en diferentes colas. Class Based Weighted Fair Queuing (CBWFQ) y Low Latency Queuing (LLQ) son ejemplos de funciones de planificación.

Criterios de selección

La tabla siguiente enumera las diferencias entre el modelado y la regulación para ayudarle a elegir la mejor solución.

	Modelado	Control de tráfico
Objetivo	Almacenar en búfer y cola los paquetes excedentes por encima de las tasas comprometidas.	Elimine (o remarque) los paquetes que exceden a las velocidades comprometidas. No almacena en búfer.
Velocidad de actualización de token	Incrementada al inicio de un intervalo de tiempo. (Se requiere una cantidad mínima de intervalos).	Continua basada en la fórmula: $1/\text{velocidad de información comprometida}$
Valores Token	Configurado en bits por segundo	Configurados en bytes.
Opciones de Configuración	<ul style="list-style-type: none"> • Comando shape en la interfaz de línea de comandos (MQC) de calidad de servicio modular para implementar el modelado basado en clase. • comando frame-relay traffic-shape para implementar el Modelado del tráfico de Frame Relay (FRTS). • comando traffic-shape para implementar el Modelado de tráfico genérico (GTS). 	<ul style="list-style-type: none"> • Comando police en el MQC para implementar políticas basadas en clase. • comando de límite de velocidad para implementar la velocidad de acceso comprometida (CAR).
Aplicable en la entrada	No	Sí

Se aplica a la salida	Sí	Sí
Ráfagas	Controla las ráfagas atenuando la velocidad de salida en intervalos de al menos ocho tiempos. Utiliza un contador dinámico para retardar el tráfico, el cual logra un efecto atenuante.	Propaga las ráfagas. No suaviza.
Ventajas	Es menos probable que los paquetes en exceso se descarten puesto que estos paquetes se almacenan en el buffer. (Paquetes de búferes hasta la longitud de la cola. Pueden producirse descartes si el tráfico en exceso se mantiene a altas velocidades.) Evita normalmente las retransmisiones debidas a paquetes descartados.	Controla la velocidad de salida a través de las caídas de los paquetes. Evita retrasos por almacenamiento en cola.
Desventajas	Puede producir retrasos debido a las colas, en especial a las colas profundas.	Descarta los paquetes en exceso (cuando está configurada), regulando los tamaños de ventana TCP y reduciendo la velocidad de salida total de los flujos de tráfico afectados. Los tamaños de ráfaga demasiado agresivos pueden llevar a una caída de paquetes excesiva y acelerar la velocidad de salir global, en particular con flujos basados en TCP.
Remarcación opcional de paquete	No	Sí (con la función CAR heredada).

* Aunque la regulación de tráfico no aplica el uso de búfers, un mecanismo de almacenamiento en cola configurado se aplica a los paquetes "aceptados" que pueden necesitar ser almacenados en cola mientras esperan ser serializados en la interfaz física.

Velocidad de actualización de token

Una diferencia fundamental entre el modelado y la regulación es la velocidad a la que se recargan los tokens. Esta sección analiza la diferencia.

Expuesto simplemente, tanto el modelado como la regulación utilizan la metáfora de cubeta con tokens. Una cubeta con ficha no posee una política de prioridad o descarte. Veamos cómo funciona la metáfora de la cubeta con ficha:

- Los token ingresan al bloque de memoria a una velocidad determinada.
- Cada token es un permiso para que el origen envíe cierto número de bits a la red.
- Para enviar un paquete, el regulador del tráfico debe ser capaz de retirar de la cubeta una cantidad de fichas equivalente al tamaño del paquete.
- Si no hay suficientes fichas en la cubeta como para enviar un paquete, el paquete espera hasta que la cubeta tenga suficientes fichas (en el caso de un modelador) o el paquete se descarta o se rebaja (en el caso de un vigilante).
- La cubeta posee una capacidad especificada. Si la capacidad del compartimiento de memoria se encuentra completa, todos los símbolos nuevos que lleguen serán descartados y no estarán disponibles para paquetes futuros. De esta manera, en cualquier momento, la ráfaga más grande que una fuente puede enviar a la red es más o menos proporcional al tamaño de la cubeta. El token bucket permite la saturación, pero la limita.

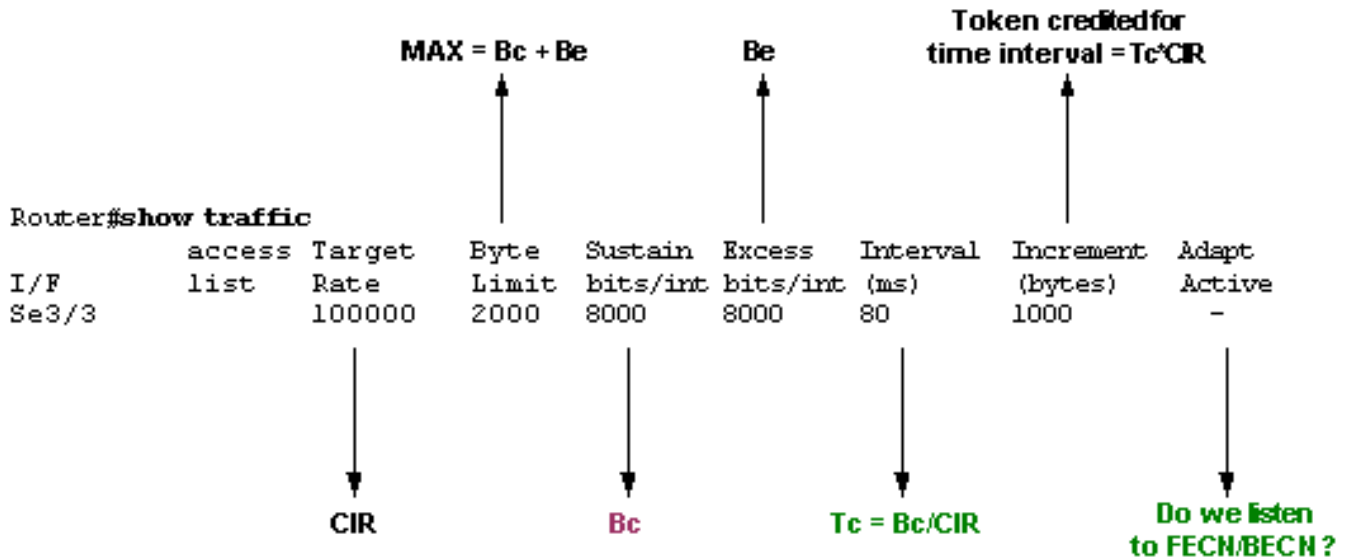
Con la cubeta con ficha metaphor en mente, veamos cómo el modelado y la regulación agregan fichas a la cubeta.

El modelado aumenta la cubeta con ficha en intervalos cronometrados mediante un valor de bits por segundo (bps). Un modelador utiliza la fórmula siguiente:

$$T_c = B_c / CIR \text{ (in seconds)}$$

En esta ecuación, B_c representa la ráfaga comprometida y CIR representa la velocidad de información comprometida. (Vea [Configuración del Modelado de Tráfico Frame Relay](#) para obtener más información.) El valor de T_c define el intervalo de tiempo durante el cual se envían los bits de B_c para mantener la tasa promedio de CIR en segundos.

El rango para T_c es de 10 a 125 ms. Con el modelado de tráfico distribuido (DTS) en Cisco 7500 Series, el T_c mínimo es de 4 ms. El router calcula internamente este valor basándose en los valores de CIR y B_c . Si el valor de B_c / CIR es menor a 125 ms, utiliza el T_c que se calcula a partir de esa ecuación. Si B_c / CIR es mayor o igual a 125 ms, usa un valor T_c interno en caso de que el IOS® de Cisco determine que el flujo de tráfico será más estable con un intervalo más reducido. Utilice el comando `show traffic-shape` para determinar si el router utiliza un valor de intervalo para T_c o el valor que se ha configurado en la línea de comandos. [El siguiente ejemplo de resultado del comando show traffic-shape se explica en comandos show para el modelado de tráfico de retransmisión de tramas.](#)



Cuando la ráfaga en exceso (Be) se configura en un valor diferente de 0, el modelador permite que los tokens se almacenen en la cubeta, hasta el valor de Bc + Be. El valor más grande que la cubeta con fichas puede llegar a alcanzar es de Bc + Be y las fichas desbordadas se pierden. La única manera para tener más que fichas Bc en la cubeta es no utilizar todas las fichas Bc durante uno o más Tc. Dado que la cubeta con tokens vuelve a cargarse todos los Tc con tokens Bc, puede acumular tokens no utilizados para usarlos posteriormente hasta Bc + Be.

En cambio, la regulación y la limitación de velocidad basadas en clases añaden tokens continuamente a la cubeta. Específicamente, la velocidad de llegada del token se calcula de la siguiente manera:

$$(time\ between\ packets < which\ is\ equal\ to\ t - t_1 > * policer\ rate) / 8\ bits\ per\ byte$$

En otras palabras, si la llegada anterior del paquete fue a t1 y la hora actual es t, la cubeta se actualiza con el valor de bytes de t-t1 basado en la velocidad de llegada del token. Observe que un regulador de tráfico utiliza valores de ráfaga especificados en bytes y la fórmula anterior convierte de bits a bytes.

Consulte un ejemplo que utiliza una CIR (velocidad de regulador) de 8000 bps y una ráfaga normal de 1000 bytes.

```
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
```

La cubeta con ficha se inicia completa a los 1000 bytes. Si llega un paquete de 450 bytes, éste se ajusta porque hay suficientes bytes en la cubeta de ficha. El paquete realiza la acción de conformidad (transmitir) y se quitan 450 bytes de la cubeta con fichas (dejando 550 bytes). Si el siguiente paquete llega 0,25 segundos más tarde, se añaden 250 bytes a la cubeta de tokens siguiendo la fórmula siguiente:

$$(0.25 * 8000) / 8$$

El cálculo deja 700 bytes en la cubeta de tokens. Si el paquete siguiente es de 800 bytes, el paquete excede el límite y se ejecuta la acción para exceso (supresión). No se toman bytes de la cubeta con ficha.

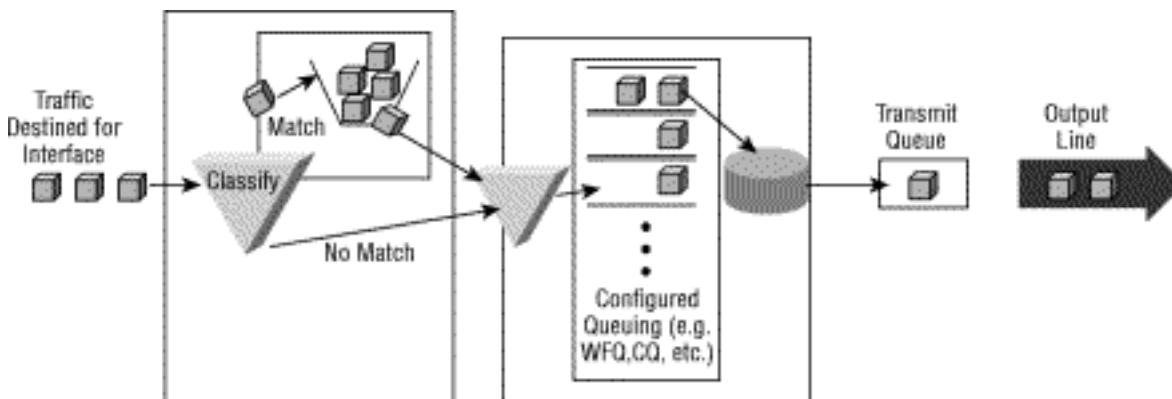
Modelado de tráfico

Cisco IOS admite los siguientes métodos de modelado de tráfico:

- [Modelado del tráfico genérico](#)
- [Diseño del Frame Relay](#)
- [Modelado basado en clase y modelado distribuido basado en clase](#)

Todos los métodos de moldeado de tráfico son similares en cuanto a la implementación, aunque sus Interfaces de línea de comandos (CLI) difieren en cierta medida y usan diferentes tipos de colas para contener y moldear el tráfico que es postergado. Cisco recomienda el modelado basado en clases y el modelado distribuido, que están configurados utilizando el CLI de Calidad de servicio (QoS) modular.

El siguiente diagrama ilustra cómo una política QoS clasifica el tráfico en clases y paquetes de colas que exceden las velocidades de formación configuradas.



[Regulación del tráfico](#)

Cisco IOS admite los siguientes métodos de regulación de tráfico:

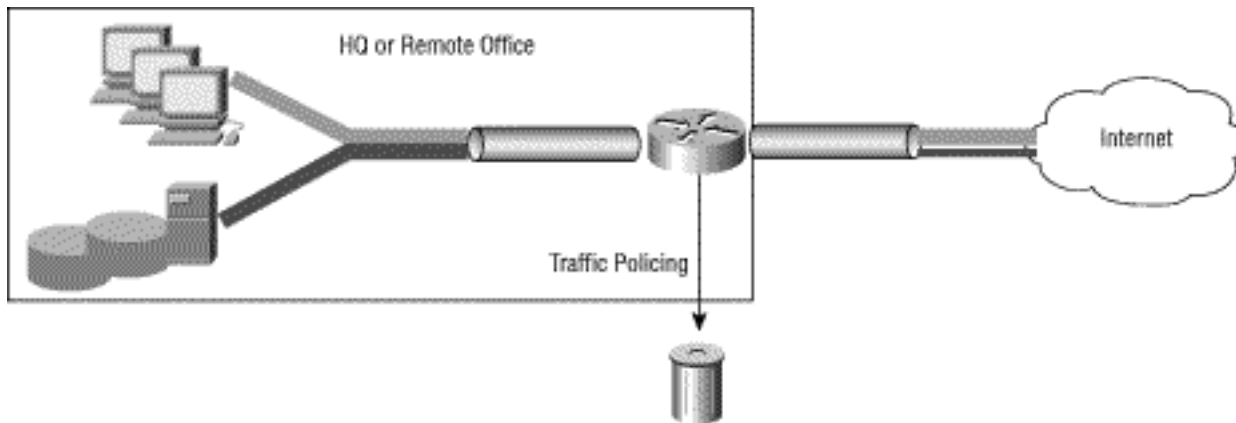
- [Velocidad comprometida de acceso](#)
- [Control basado en clase](#)

Los dos mecanismos tienen diferencias funcionales importantes, como se explica en [Comparación de las políticas en base a la clase y velocidad de acceso comprometido](#). Cisco recomienda la regulación basada en clases y otras funciones de la CLI QoS modular cuando se apliquen políticas QoS.

Utilice el comando **police** para especificar que una clase de tráfico debe tener una velocidad máxima impuesta y, si se excede esa velocidad, debe emprenderse una acción inmediata. En otras palabras, con el comando **police** no existe la opción de almacenar el paquete en el búfer y enviarlo más tarde, como sucede con el comando **shape**.

Además, con la regulación de tráfico, la cubeta con ficha determina si el paquete excede la velocidad aplicada o se ajusta a ella. En todo caso, la regulación de tráfico implementa una acción configurable que incluye la configuración de la precedencia de IP o punto de código de servicios diferenciados (DSCP).

El siguiente diagrama ilustra una aplicación común de la regulación del tráfico en un punto de gestión, donde generalmente se aplican características de QoS.



Controles de ancho de banda mínimo versus máximo

Los comandos `shape` y `police`, restringen la velocidad de salida a un valor máximo en kbps. Cabe destacar que ningún mecanismo provee una garantía de ancho de banda mínimo durante periodos de congestión. Utilice los comandos `priority` o `bandwidth` para proveer tales garantías.

Una política jerárquica utiliza dos políticas de servicio: una política principal para aplicar un mecanismo QoS a un tráfico total y una política secundaria para aplicar un mecanismo QoS a un flujo o subconjunto del tráfico total. Las interfaces lógicas, como subinterfaces y interfaces de túnel, requieren una política jerárquica con la función de limitación del tráfico en el nivel principal y el envío a cola en los niveles inferiores. La función de limitación del tráfico reduce la velocidad de salida y (probablemente) crea congestión, desde el punto de vista de los paquetes en exceso en cola.

La configuración siguiente es subóptima y se muestra para ilustrar la diferencia entre el comando **police** y el comando `shape` cuando se limita un tráfico total, en este caso de clases predeterminadas, a una velocidad máxima. En esta configuración, el comando **police** envía los paquetes de las clases secundarias basándose en el tamaño del paquete y el número de bytes que quedan en las cubetas de tokens conformes y en exceso. (Vea [Regulación del Tráfico](#).) El resultado es que las velocidades asignadas a las clases Voz sobre IP (VoIP) y Protocolo de Internet (IP) no pueden garantizarse puesto que la **función de regulación** invalida las garantías ofrecidas por la función de prioridad.

Sin embargo, si se utiliza el comando `shape`, el resultado es un sistema jerárquico de colocación de cola y se realizan todas las garantías. En otras palabras, cuando la carga ofrecida excede la velocidad modelada, a las clases de IP y VoIP se les garantiza su velocidad y el tráfico de clase predeterminada (en el nivel secundario) sufre las pérdidas.

Precaución: No se recomienda esta configuración y se muestra para ilustrar la diferencia entre el comando `police` versus `shape` cuando se limita el tráfico global.

```
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 la configuración anterior tenga sentido, la regulación debe sustituirse por el modelado.
Por ejemplo:

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

Para aprender más sobre el padre y las políticas hija, refiera por favor a la [política de servicio del niño de QoS para la clase de prioridad](#).

[Información Relacionada](#)

- [Soporte de la Tecnología QoS](#)
- [Soporte Técnico - Cisco Systems](#)