

Traffic Policy en verkeersvorm vergelijken om bandbreedte te beperken

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Conventies](#)

[Gebruikte componenten](#)

[Achtergrondinformatie](#)

[Toezicht versus vormgeving](#)

[Selectiecriteria](#)

[Token Refresh Rate](#)

[Traffic Shaping](#)

[Traffic policing](#)

[Minimale versus maximale bandbreedtecontroles](#)

[Gerelateerde informatie](#)

Inleiding

Dit document beschrijft de functionele verschillen tussen traffic shaping en traffic policing, die beide de uitvoersnelheid beperken.

Voorwaarden

Vereisten

Er zijn geen specifieke vereisten van toepassing op dit document.

Conventies

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

Gebruikte componenten

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

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u zorgen dat u de potentiële impact van elke opdracht begrijpt.

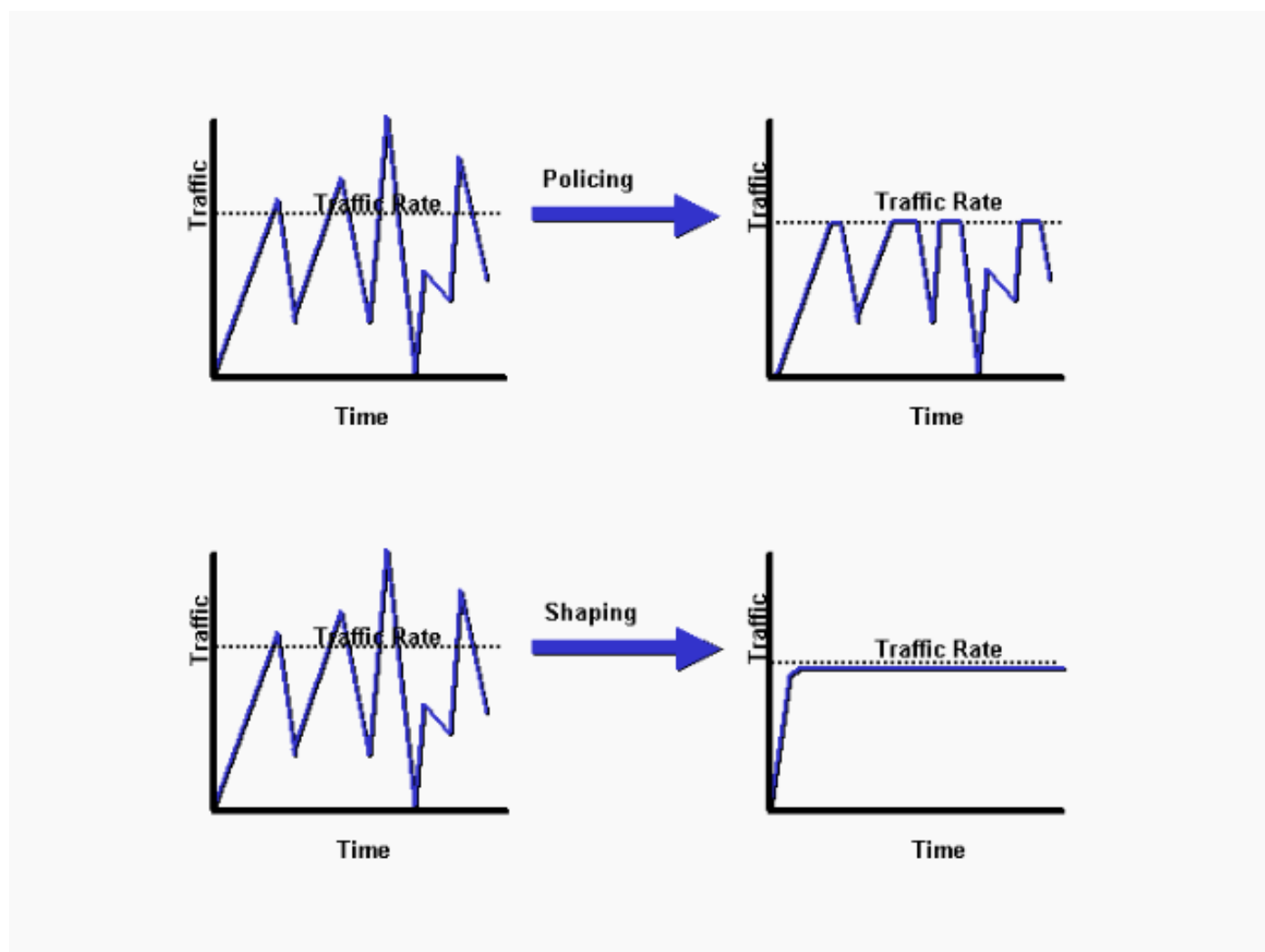
Achtergrondinformatie

Dit document verduidelijkt de functionele verschillen tussen traffic shaping en toezicht. Allebei beperken functioneel het tarief van de verkeersoutput. Beide mechanismen gebruiken een symbolische emmer als een verkeersmeter om de pakketsnelheid te meten. Zie [Wat is een Token Bucket voor](#) meer informatie over [symbolische emmers?](#)

Toezicht versus vormgeving

Traffic policing propageert uitbarstingen. Wanneer het verkeerstarief het ingestelde maximumtarief bereikt, wordt overtollig verkeer verbroken (of gemerkt). Het resultaat is een outputsnelheid die verschijnt als een zaagtand met toppen en troggen. In tegenstelling tot het toezicht, behoudt traffic shaping overtollige pakketten in een wachtrij en plant vervolgens het overtollige voor latere transmissie over perioden. Het resultaat van traffic shaping is een vloeiende uitvoersnelheid van het pakket.

In het volgende diagram worden de belangrijkste verschillen tussen de twee verkeersopties weergegeven.



Policing VS Shaping

Shaping impliceert het bestaan van een wachtrij en van voldoende geheugen om vertraagde pakketten te bufferen, terwijl toezicht dat niet doet. Wachtrijen zijn een uitgaand concept. pakketten die een interface verlaten, worden in de wachtrij geplaatst en kunnen worden gevormd.

Alleen toezicht kan worden toegepast op inkomend verkeer op een interface. Zorg ervoor dat u voldoende geheugen hebt wanneer u het vormen toelaat. Bovendien vereist het vormen een functie die voor recentere transmissie van om het even welke vertraagde pakketten plant. Met deze planningsfunctionaliteit kunt u de vormwachtrij in verschillende wachtrijen indelen. Voorbeelden van deze functionaliteit zijn Class Based Weighted Fair Queuing (CBWFQ) en lage latentie Queuing (LLQ).

Selectiecriteria

De volgende tabel toont de verschillen tussen vormgeving en toezicht om u te helpen de juiste verkeersoplossing te kiezen.

	Vormgeving	Toezicht
Doel	Buffer en wachtrij van overtollige pakketten via de vastgelegde tarieven.	Overtollige pakketten laten vallen (of opmerken) op de vastgelegde tarieven. Buffert niet.*
Token Refresh Rate	Verhoogd aan het begin van een tijdsinterval. (Het minimumaantal tussenpozen is vereist.)	Continu gebaseerd op formule: $1 / \text{gecommiteerde informatiesnelheid}$
Token waarden	Geconfigureerd in bits per seconde.	Ingesteld in bytes.
Configuratieopties	<ul style="list-style-type: none"> • FormFiller-opdracht in de modulaire Quality of Service Command-Line Interface (MQC) om op klasse gebaseerde shaping te implementeren. • Frame Relay Traffic Shaping-opdracht voor implementatie van Frame Relay Traffic Shaping (FRTS). • Hiermee implementeert u Generic Traffic Shaping (GTS). 	<ul style="list-style-type: none"> • het bevel van de politie in de MQC om op klasse gebaseerde politieactiviteiten uit te voeren. • tarieflijm-opdracht om gecommiteerde toegangstarief te implementeren (CAR).
Van toepassing op inkomende services	Nee	Ja
Van toepassing op uitgaand	Ja	Ja
Uitbarstingen	Bestuurt barsten en vereffent de uitvoersnelheid over minimaal acht tijdsintervallen. Maakt gebruik van een lekke emmer om verkeer uit te stellen, waardoor een vloeiend effect wordt bereikt. Het is minder waarschijnlijk dat u overtollige pakketten laat vallen omdat overtollige pakketten worden gebufferd. (Buffers pakketten tot de lengte van de wachtrij. Er kunnen druppels optreden als er veel te veel verkeer is.) Vermijd normaal heruitzendingen als gevolg van gedropte pakketten.	Verspreidt uitbarstingen. Vloeiend maken niet mogelijk.
Voordelen		Bestuurt de uitvoersnelheid door pakketdalingen. Vermijd vertragingen als gevolg van queuing.

Nadelen	Kan vertraging introduceren door queuing, met name diepe rijen.	Laat overtollige pakketten vallen (wanneer geconfigureerd), gethrottled TCP-venstergrootte en verlaagt de algemene uitvoersnelheid van getroffen verkeersstromen. De overmatig agressieve burst grootte kan tot bovenmatige pakketdalingen leiden het algemene outputtarief, in het bijzonder met op gebaseerde stromen verstikken.
Optionele pakketmarkerin g	Nee	Ja (met verouderde CAR-functie).

* Hoewel de controle geen buffer toepast, een geconfigureerd queuing Het mechanisme is op **conformed** pakketten van toepassing die kunnen worden een rij gevormd terwijl zij wachten om bij de fysieke interface worden in series vervaardigd.

Token Refresh Rate

Een belangrijk verschil tussen vormgeven en toezicht is de snelheid waarmee tokens worden aangevuld. Zowel vormgeven als toezicht maken gebruik van de symboolmetafoor. Een symbolische emmer heeft zelf geen teruggooi of prioritair beleid.

Met token emmer functionaliteit:

- Tokens worden met een bepaalde snelheid in de emmer gestopt.
- Elk token is toestemming voor de bron om een bepaald aantal bits naar het netwerk te sturen.
- Om een pakket te verzenden, moet de verkeersregelaar in staat zijn om een aantal tokens te verwijderen dat gelijk is aan de pakketgrootte.
- Als er niet genoeg tokens in de emmer zitten om een pakket te versturen, wacht het pakket ofwel tot de emmer genoeg tokens heeft (in het geval van een shaper) of het pakket wordt weggegooid of afgezet (in het geval van een policer).
- De emmer zelf heeft een specifieke capaciteit. Als de emmer vult tot capaciteit, worden nieuwe tokens die aankomen weggegooid en zijn niet beschikbaar voor toekomstige pakketten. Dus op elk moment is de grootste uitbarsting die een bron naar het netwerk kan sturen ruwweg evenredig met de grootte van de emmer. Een symbolische emmer maakt barsttheid mogelijk, maar beperkt het.

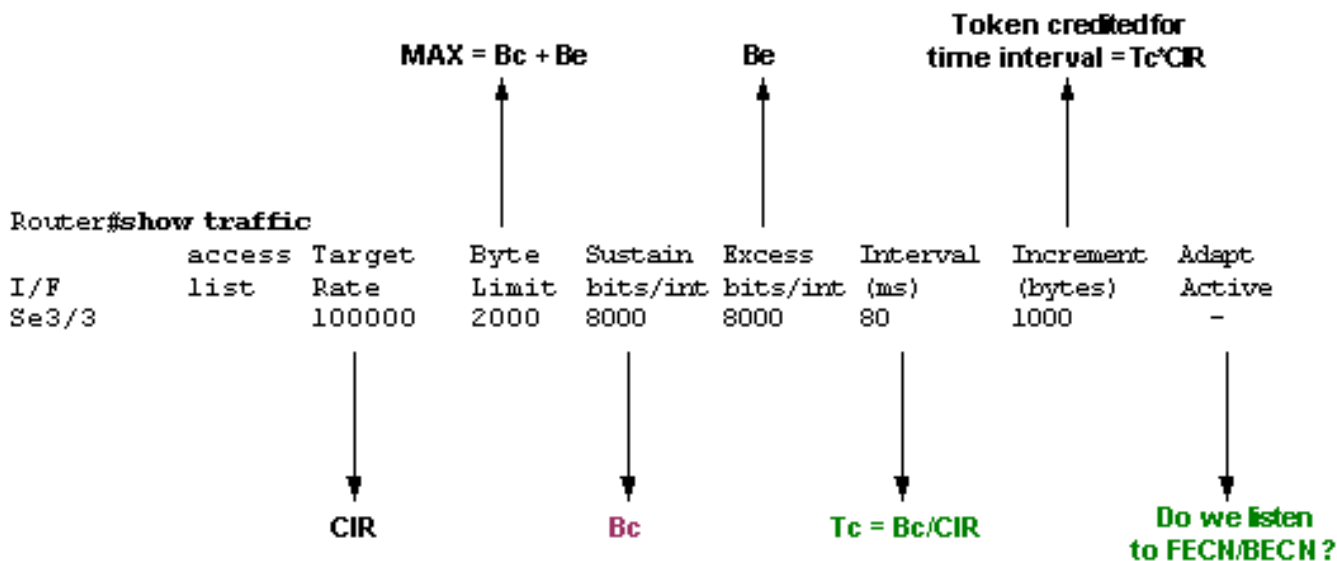
De vormende toename van de symbolische emmer met vastgestelde intervallen die een beetje per seconde (bps) waarde gebruiken. Een shaper gebruikt de volgende formule:

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

In deze vergelijking geeft B_c de toegezegde uitbarsting weer, en CIR staat voor toegezegde informatiesnelheid. (Zie [Frame Relay Traffic Shaping configureren](#) voor meer informatie.) De waarde van T_c bepaalt het tijdsinterval waarin u de B_c -bits verzendt om het gemiddelde tarief van de CIR in seconden te handhaven.

Het bereik voor T_c ligt tussen 10 ms en 125 ms. Met Distributed Traffic Shaping (DTS) op Cisco 7500 Series is de minimum-TAC 4 ms. De router berekent deze waarde intern op basis van de CIR- en B_c -waarden. Als B_c/CIR minder dan 125 ms bedraagt, wordt de met die vergelijking berekende T_c gebruikt. Als B_c/CIR 125 ms of meer bedraagt, wordt een interne T_c -waarde

gebruikt als Cisco IOS[®] bepaalt dat de verkeersstroom met een kleiner interval stabiel kan zijn. Gebruik het bevel van de **show verkeer-vorm** om te bepalen of uw router een interne waarde voor Tc of de waarde gebruikt die u bij de bevel-lijn vormde. De volgende voorbeelduitvoer van de opdracht **show traffic-form** wordt uitgelegd in [Show Commands for Frame Relay Traffic Shaping](#).



toon verkeersoutput

Wanneer de overtollige barst (Be) is ingesteld op een andere waarde dan 0, kunnen tokens in de emmer worden opgeslagen, tot Bc + Be. De grootste waarde die de token emmer ooit kan bereiken is Bc + Be, en overflow tokens worden gedropt. De enige manier om meer dan Bc tokens in de emmer te hebben is niet alle Bc tokens te gebruiken tijdens een of meer Tc. Aangezien de token bucket elke Tc met Bc tokens wordt gevuld, kunt u ongebruikte tokens verzamelen voor later gebruik tot Bc + Be.

In tegenstelling, op klasse gebaseerde toezicht en tarief-limiting voegt continu tokens aan de emmer toe. Met name wordt de symbolische aankomstnelheid als volgt berekend:

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

Met andere woorden, als de vorige aankomst van het pakket op t1 was en de huidige tijd t is, wordt de emmer bijgewerkt met t-t1 waarde van bytes die op de symbolische aankomstnelheid worden gebaseerd. Merk op dat een traffic policer burst-waarden gebruikt die in bytes zijn gespecificeerd, en dat de vorige formule van bits naar bytes converteert.

Dit is een voorbeeld waarin een CIR (of policer rate) van 8000 bps en een normale burst van 1000 bytes wordt gebruikt:

```
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
```

De symbolische emmers beginnen volledig op 1000 bytes. Als een pakket van 450 bytes wordt ontvangen, is het pakket conform omdat er voldoende bytes beschikbaar zijn in de token-emmer. De conforme actie (zend) wordt door het pakket genomen en 450 bytes worden verwijderd uit de token bucket (en 550 bytes achterlaten). Als het volgende pakket .25 seconden later arriveert, worden 250 bytes aan de token bucket toegevoegd volgens de volgende formule:

(0.25 * 8000) / 8

De berekening laat 700 bytes in de token emmer. Als het volgende pakket 800 bytes is, overschrijdt het pakket en overschrijdt de actie (daling) wordt genomen. Er worden geen bytes uit de token bucket gehaald.

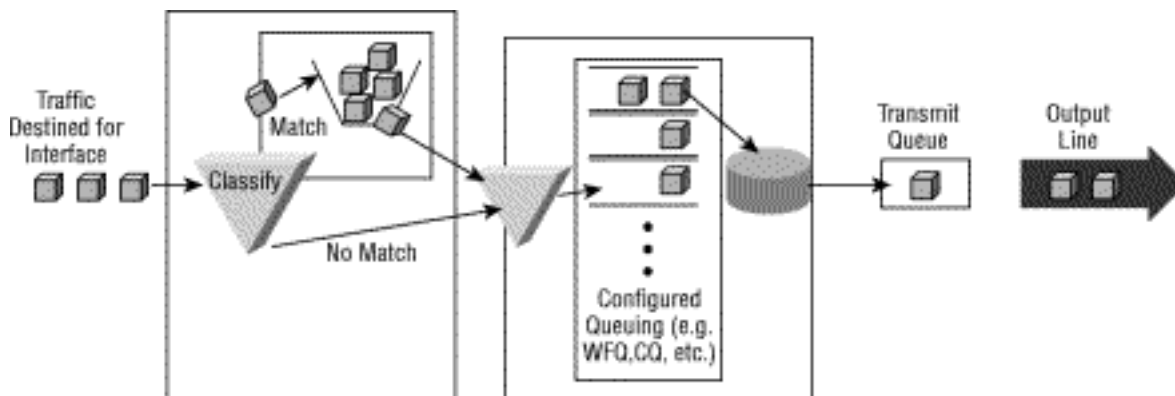
Traffic Shaping

Cisco IOS ondersteunt de volgende methoden voor traffic shaping:

- [Generic Traffic Shaping](#)
- [Frame Relay Traffic Shaping](#)
- [Op klasse gebaseerde shaping en gedistribueerde op klasse gebaseerde shaping](#)

Alle traffic shaping methoden zijn vergelijkbaar in implementatie, hoewel hun opdrachtregelinterfaces (CLI's) enigszins verschillen en ze verschillende typen wachtrijen gebruiken om verkeer te bevatten en te vormen dat wordt uitgesteld. Cisco raadt op klasse gebaseerde shaping en gedistribueerde shaping aan, die zijn geconfigureerd met de modulaire QoS CLI.

Het volgende diagram illustreert hoe een QoS-beleid verkeer in klassen ordent en pakketten in wachtrijen plaatst die de ingestelde vormtarieven overschrijden.



Traffic policing

Cisco IOS ondersteunt de volgende methoden van traffic policing:

- [Toegewezen toegangssnelheid](#)
- [Op klasse gebaseerde toezicht](#)

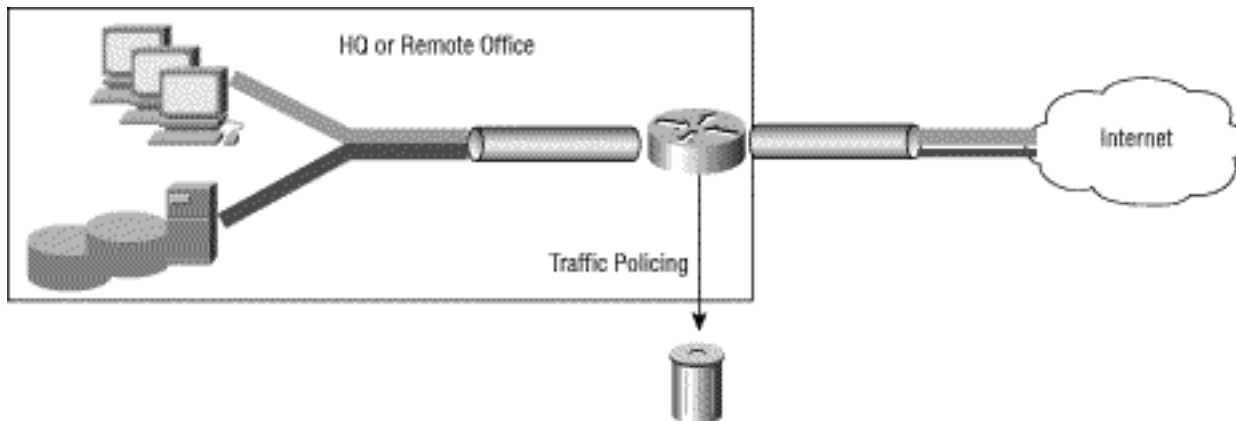
De twee mechanismen hebben belangrijke functionele verschillen, zoals wordt uitgelegd in [Compare Class-Based Policing en Committed Access Rate](#). Cisco raadt op klasse gebaseerde toezicht en andere functies van de modulaire QoS-CLI aan wanneer QoS-beleid wordt toegepast.

Gebruik het politiebevel om aan te geven dat een bepaalde verkeersklasse een maximumtarief moet hebben en dat bij overschrijding daarvan onmiddellijk maatregelen moeten worden genomen. Met andere woorden, met het commando van de **politie** is het geen optie om het pakket te bufferen en het later te versturen, zoals het geval is met het bevel van de **vorm**.

Daarnaast bepaalt de token bucket met policing of een pakket de toegepaste snelheid overschrijdt of hiermee in overeenstemming is. In beide gevallen implementeert het toezicht een

configureerbare actie, die de IP-voorrang of het gedifferentieerde servicescodepunt (DSCP) omvat.

Het volgende diagram illustreert een gemeenschappelijke toepassing van traffic policing op een congestiepunt, waar QoS-functies over het algemeen van toepassing zijn.



Minimale versus maximale bandbreedtecontroles

Zowel de commando's voor **vorm** als **politie** beperken de uitvoersnelheid tot een maximale kbps waarde. Belangrijk is dat geen van beide mechanismen een minimale bandbreedtegarantie biedt in perioden van stremming. Gebruik de **bandbreedte** of **prioriteitsopdracht** om dergelijke garanties te bieden.

Een hiërarchisch beleid gebruikt twee dienstenbeleid - een ouderbeleid om een mechanisme QoS op een verkeersaggregaat en een kindbeleid toe te passen om een mechanisme QoS op een stroom of een ondergroep van het aggregaat toe te passen. Logische interfaces, zoals subinterfaces en tunnelinterfaces, vereisen een hiërarchisch beleid met de traffic pointslimiting en wachtrijen op lagere niveaus. Het verkeer-limiting functie verlaagt de uitvoersnelheid en leidt (vermoedelijk) tot congestie, zoals gezien door queuing extra pakketten.

De volgende configuratie is niet optimaal en wordt getoond om het verschil te illustreren tussen de **politie** versus het **vormig** bevel wanneer limiting een verkeersaggregaat, in dit geval class-default, tot een maximumtarief. In deze configuratie stuurt het **politiebevel** pakketten vanuit de kindklassen op basis van de grootte van het pakket en het aantal bytes dat in de conforme band blijft en de symbolische emmers overschrijdt. (Zie [Traffic policing](#).) Het gevolg is dat de tarieven die worden gegeven aan de klassen Voice over IP (VoIP) en Internet Protocol (IP) niet kunnen worden gegarandeerd, aangezien de **politie**-functie de garanties die door de **prioritaire** functie worden gegeven, overtreedt.

Echter, als de **vorm** commando wordt gebruikt, is het resultaat een hiërarchische wachtrij systeem, en alle garanties zijn gemaakt. Met andere woorden, wanneer de aangeboden belasting de vormsnelheid overschrijdt, zijn de VoIP- en IP-klassen verzekerd van hun snelheid, en het class-default verkeer (op het kinderniveau) ondergaat elke daling.

Voorzichtig: Deze configuratie wordt niet aanbevolen en wordt getoond om het verschil te illustreren tussen de **politie** versus het **form** commando wanneer het een verkeersaggregaat beperkt.

```
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
```

Om de vorige configuratie zin te geven, moet het toezicht worden vervangen door vormen.
Voorbeeld:

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

Opmerking: Raadpleeg [QoS-kinderservicebeleid voor prioriteitsklasse voor](#) meer informatie over ouder- en [kinderbeleid](#).

Gerelateerde informatie

- [QoS-technologieondersteuning \(Quality of Services\)](#)
- [Technische ondersteuning en documentatie – Cisco Systems](#)

Over deze vertaling

Cisco heeft dit document vertaald via een combinatie van machine- en menselijke technologie om onze gebruikers wereldwijd ondersteuningscontent te bieden in hun eigen taal. Houd er rekening mee dat zelfs de beste machinevertaling niet net zo nauwkeurig is als die van een professionele vertaler. Cisco Systems, Inc. is niet aansprakelijk voor de nauwkeurigheid van deze vertalingen en raadt aan altijd het oorspronkelijke Engelstalige document ([link](#)) te raadplegen.