

# Modos do Tunelamento do DiffServ para redes MPLS

## Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenções](#)

[Informações de Apoio](#)

[Comportamento padrão](#)

[Pre comando usage e comportamento do Cisco IOS Software 12.2\(13\)T](#)

[Comando usage e comportamento do Cisco IOS Software 12.2\(13\)T do cargo](#)

[Modos do Tunelamento do DiffServ](#)

[Modo uniforme](#)

[Modo da tubulação](#)

[Modo da Curto-tubulação](#)

[Sumário do modo de túnel](#)

[Informações Relacionadas](#)

## [Introdução](#)

Este documento descreve a implantação dos modos de tunelamento de Differentiated Services (Diffserv) disponíveis para ambientes de rede baseados em Multiprotocol Label Switching (MPLS).

## [Pré-requisitos](#)

### [Requisitos](#)

Os leitores deste documento devem estar cientes destes tópicos:

- MPLS e MPLS para o Virtual Private Networks (VPN)
- Conceitos em relação à Precedência IP, ao Tipo de serviço (ToS) e ao DiffServ
- Quality of Service (QoS) Packet Marking e classificação usando a interface de linha CLI do comando modular qos (MQC)

### [Componentes Utilizados](#)

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

- Liberação 12.2(13)T do <sup>® do</sup> Cisco IOS Software para características de QoS novas. A liberação 12.1(5)T inclui características de QoS originais.
- Algum roteador Cisco do 3600 Series ou de mais tarde, como o Cisco 3660 ou os 7206 que apoia o núcleo do fornecedor MPLS (P) funcionalidade do roteador do roteador/ponta de provedor (PE).

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.

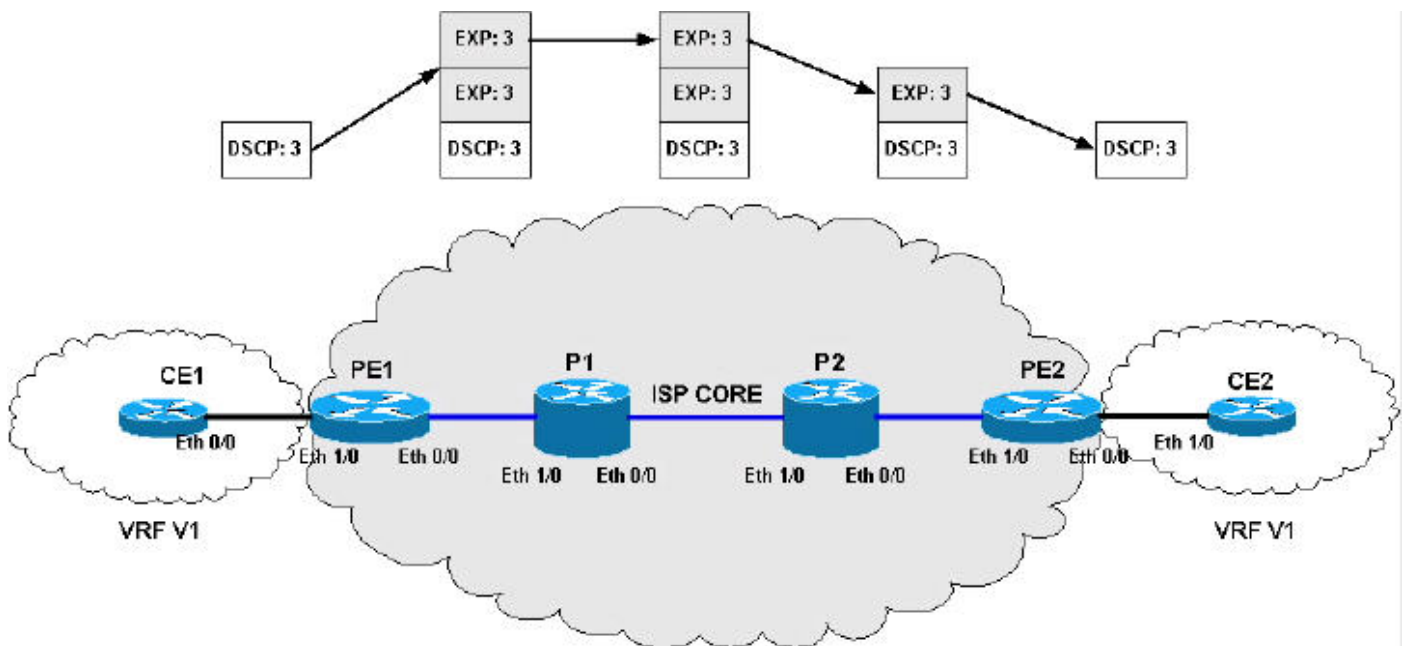
## Convenções

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

## Informações de Apoio

### Comportamento padrão

Este diagrama descreve o comportamento padrão dos bit experimentais do DiffServ Code Point (DSCP) /MPLS (EXP) enquanto um pacote viaja de um roteador do edge de cliente (CE) a um outro CE Router através de um núcleo MPLS:



Este restante desta seção descreve a atividade dentro do diagrama do comportamento padrão.

Imposição da etiqueta (IP - > etiqueta):

- A Precedência IP do pacote do IP recebido é copiada aos bit MPLS EXP de todas as etiquetas empurradas.
- Os três bits do bit DSCP são copiados aos bit MPLS EXP de todas as etiquetas empurradas.
- Esta técnica é sabida igualmente como a reflexão de ToS.

Transmissão MPLS (etiqueta – > etiqueta):

- O EXP é copiado às etiquetas novas que são trocadas/empurradas durante a transmissão ou a imposição.
- Na imposição de rótulo, as etiquetas sendo a base não são alteradas com o valor da etiqueta nova que está sendo adicionada à pilha de rótulo atual.
- Na disposição de rótulo, os bit EXP não são copiados para baixo aos bit recentemente expostos da etiqueta EXP.

Disposição da etiqueta (etiqueta – > IP):

- Na disposição de rótulo os bit EXP não são copiados para baixo ao campo IP precedence/DSCP do pacote IP recentemente exposto.

## [Pre comando usage e comportamento do Cisco IOS Software 12.2\(13\)T](#)

Antes da versão do IOS 12.2(13)T, o [comando set mpls experimental](#) era o único método disponível para alterar os bit MPLS EXP.

## [Comando usage e comportamento do Cisco IOS Software 12.2\(13\)T do cargo](#)

Com versão do IOS 12.2(13)T e mais tarde, o [comando set mpls experimental](#) é alterado permitir estas opções:

- [ajuste o MPLS Experimental o mais elevado {mpls-Exp-valor / \[table table-map-name\] do qos-grupo}](#)
- [ajuste a imposição do MPLS Experimental {mpls-Exp-valor / \[table table-map-name\] do qos-grupo}](#)

**Note:** O comando [set mpls experimental topmost](#) novo é equivalente ao comando [set mpls imposition](#) idoso.

Estes dois comandos, em combinação com algum Switches do comando [new](#), permitem o melhor controle da manipulação de bit MPLS EXP durante o impulso, a troca, e as operações POP da etiqueta. Estes dois comandos permitem que você use modos do Tunelamento do DiffServ.

## [Modos do Tunelamento do DiffServ](#)

Os modos do Tunelamento do DiffServ introduzem um Per-Hop Behavior novo (PHB), que permite QoS diferenciado em uma rede de fornecedores. O modo do Tunelamento é definido na borda da rede, normalmente no Label Switch Routers PE (LSR) (ingresso e saída). Você pode precisar de fazer mudanças no Roteadores P; você deve igualmente considerar o que ocorre quando o rótulo mais superior é removido de um pacote devido ao Penultimate Hop Popping (PHP). Pode ser necessário copiar o valor MPLS EXP da etiqueta superior que está sendo estalada à etiqueta recentemente exposta; isto não se aplica sempre a todos os modos do Tunelamento.

Em alguns casos (por exemplo, uma rede MPLS NON-VPN lisa), a ação PHP no roteador final P pode expor um pacote IP liso quando um pacote com somente uma etiqueta é recebido. Quando este pacote IP estiver recebido pela saída LSR (PE), não é possível classificar o pacote baseado nos bit MPLS EXP porque não há nenhuma etiqueta agora. Nestas situações, você deve

configurar o roteador de PE da saída para anunciar um **rótulo null explícito**. Quando a ação PHP é executada no roteador P, uma etiqueta com um valor de zero está enviada, e por meio de esta etiqueta especial você pode identificar os bit EXP como normalmente pacotes rotulados, permitindo a classificação correta no roteador de PE da saída.

O suporte de rede MPLS à especificação Diffserv define estes modos do Tunelamento:

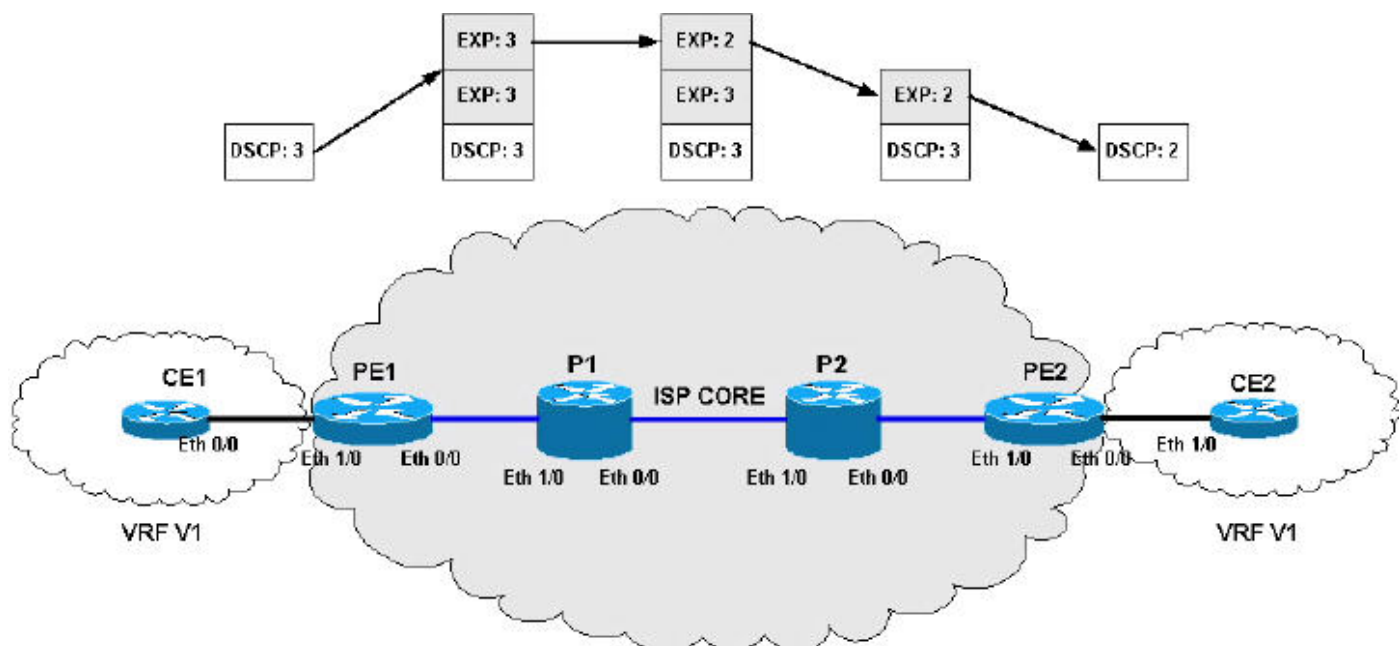
- [Uniforme](#)
- [Tubulação](#)
- [Curto-tubulação](#)

As próximas seções examinam cada modo do Tunelamento separadamente, e fornecem exemplos para mostrar como cada modo do Tunelamento pode ser configurado. Os exemplos incluem um mapeamento completo da Precedência IP aos bit MPLS EXP. É possível ter um número parâmetros de QoS e de modos diferentes do Tunelamento para cada cliente.

**Note:** Os exemplos de configuração não são específicos para o MPLS VPN, e são aplicáveis para redes lisas das redes MPLS e do CARRIER supported CARRIER (CSC). É igualmente possível que sua rede pode variar de uma outra rede — muitos parâmetros de QoS e modos diferentes do Tunelamento podem ser usados.

## Modo uniforme

O modo uniforme do Tunelamento do DiffServ tem somente uma camada de QoS, que alcança fim-a-fim. O roteador de PE do ingresso (PE1) copia o DSCP do pacote do IP recebido nos bit MPLS EXP das etiquetas impostas. Enquanto os bit EXP viajam com o núcleo, podem ou não serem alterados pelo Roteadores intermediário P. Neste exemplo, o roteador P1 altera os bit EXP da etiqueta superior. No roteador da saída P (P2) nós copiamos os bit EXP aos bit EXP da etiqueta recentemente exposta após o PHP (Penúltimo-Salto-PNF). Finalmente no roteador de PE da saída (PE2) nós copiamos então os bit EXP aos bit DSCP do pacote IP recentemente exposto.



Configurações do modo uniforme:

**PE1**

```
!--- This configuration maps the IP Precedence !--- of
the incoming IP packets to the MPLS EXP bits. class-map
match-all IP-AF11 match ip precedence 0 class-map match-
all IP-AF12 match ip precedence 1 class-map match-all
IP-AF21 match ip precedence 2 class-map match-all IP-
AF22 match ip precedence 3 class-map match-all IP-AF31
match ip precedence 4 class-map match-all IP-AF32 match
ip precedence 5 class-map match-all MPLS-AF11 match mpls
experimental topmost 0 class-map match-all MPLS-AF12
match mpls experimental topmost 1 class-map match-all
MPLS-AF21 match mpls experimental topmost 2 class-map
match-all MPLS-AF22 match mpls experimental topmost 3
class-map match-all MPLS-AF31 match mpls experimental
topmost 4 class-map match-all MPLS-AF32 match mpls
experimental topmost 5 policy-map set-MPLS-PHB
  class IP-AF11
    police 8000 conform-action set-mpls-exp-imposition-
transmit
                                0 exceed-
action drop
  class IP-AF12
    police 10000 conform-action set-mpls-exp-imposition-
transmit
                                1 exceed-action set-mpls-exp-imposition-
transmit 0
  class IP-AF21
    police 12000 conform-action set-mpls-exp-imposition-
transmit
                                2 exceed-action set-mpls-exp-imposition-
transmit 1
  class IP-AF22
    police 12000 conform-action set-mpls-exp-imposition-
transmit
                                3 exceed-action set-mpls-exp-imposition-
transmit 2
  class IP-AF31
    police 12000 conform-action set-mpls-exp-imposition-
transmit
                                4 exceed-action set-mpls-exp-imposition-
transmit 3
  class IP-AF32
    police 12000 conform-action set-mpls-exp-imposition-
transmit
                                5 exceed-action set-mpls-exp-imposition-
transmit 4
```

**policy-map output-qos**

```
class MPLS-AF11
  bandwidth percent 5
  random-detect
class MPLS-AF12
  bandwidth percent 10
  random-detect
class MPLS-AF21
  bandwidth percent 10
  random-detect
class MPLS-AF22
  bandwidth percent 15
  random-detect
class MPLS-AF31
  bandwidth percent 20
```

```

    random-detect
class MPLS-AF32
    bandwidth percent 30
    random-detect

interface Ethernet0/0
    ip address 192.168.0.1 255.255.255.0
    max-reserved-bandwidth 90
    service-policy output output-qos
    tag-switching ip
!
interface Ethernet1/0
    ip vrf forwarding v1
    ip address 10.0.0.2 255.255.255.0
    service-policy input set-MPLS-PHB
!

```

## P1

```

!--- This configuration swaps the top label from 3 to 2,
!--- which does not need to occur to follow the previous
configuration. class-map match-all mpls-in match mpls
experimental topmost 3 ! policy-map mpls-in
    class mpls-in
        set mpls experimental topmost 2
!
interface Ethernet0/0
    ip address 192.168.1.1 255.255.255.0
    tag-switching ip
!
interface Ethernet1/0
    ip address 192.168.0.2 255.255.255.0
    service-policy input mpls-in
    tag-switching ip
!

```

## P2

```

!--- Remember to copy down the MPLS EXP value !--- from
the newly exposed label after the PHP. class-map match-
all MPLS-AF11 match mpls experimental topmost 0 class-
map match-all MPLS-AF12 match mpls experimental topmost
1 class-map match-all MPLS-AF21 match mpls experimental
topmost 2 class-map match-all MPLS-AF22 match mpls
experimental topmost 3 class-map match-all MPLS-AF31
match mpls experimental topmost 4 class-map match-all
MPLS-AF32 match mpls experimental topmost 5 ! class-map
match-all qos-group-AF11 match qos-group 0 class-map
match-all qos-group-AF12 match qos-group 1 class-map
match-all qos-group-AF21 match qos-group 2 class-map
match-all qos-group-AF22 match qos-group 3 class-map
match-all qos-group-AF31 match qos-group 4 class-map
match-all qos-group-AF32 match qos-group 5 ! policy-map
qos-group-in
    class MPLS-AF11
        set qos-group mpls experimental topmost
    class MPLS-AF12
        set qos-group mpls experimental topmost
    class MPLS-AF21
        set qos-group mpls experimental topmost
    class MPLS-AF22

```

```

    set qos-group mpls experimental topmost
class MPLS-AF31
    set qos-group mpls experimental topmost
class MPLS-AF32
    set qos-group mpls experimental topmost
!
policy-map qos-group-out
class qos-group-AF11
    bandwidth percent 5
    random-detect
    set mpls experimental topmost qos-group
class qos-group-AF12
    bandwidth percent 10
    random-detect
    set mpls experimental topmost qos-group
class qos-group-AF21
    bandwidth percent 10
    random-detect
    set mpls experimental topmost qos-group
class qos-group-AF22
    bandwidth percent 15
    random-detect
    set mpls experimental topmost qos-group
class qos-group-AF31
    bandwidth percent 20
    random-detect
    set mpls experimental topmost qos-group
class qos-group-AF32
    bandwidth percent 30
    random-detect
    set mpls experimental topmost qos-group
!
interface Ethernet0/0
    ip address 192.168.2.1 255.255.255.0
    max-reserved-bandwidth 90
service-policy output qos-group-out
    tag-switching ip
!
interface Ethernet1/0
    ip address 192.168.1.2 255.255.255.0
service-policy input qos-group-in
    tag-switching ip
!

```

## PE2

```

!--- Remember to copy down the MPLS EXP bits to the IP
Precedence !--- of the newly exposed IP packet. class-
map match-all MPLS-AF11 match mpls experimental topmost
0 class-map match-all MPLS-AF12 match mpls experimental
topmost 1 class-map match-all MPLS-AF21 match mpls
experimental topmost 2 class-map match-all MPLS-AF22
match mpls experimental topmost 3 class-map match-all
MPLS-AF31 match mpls experimental topmost 4 class-map
match-all MPLS-AF32 match mpls experimental topmost 5 !
class-map match-all qos-group-AF11 match qos-group 0
class-map match-all qos-group-AF12 match qos-group 1
class-map match-all qos-group-AF21 match qos-group 2
class-map match-all qos-group-AF22 match qos-group 3
class-map match-all qos-group-AF31 match qos-group 4
class-map match-all qos-group-AF32 match qos-group 5 !
policy-map qos-group-in

```

```

class MPLS-AF11
  set qos-group mpls experimental topmost
class MPLS-AF12
  set qos-group mpls experimental topmost
class MPLS-AF21
  set qos-group mpls experimental topmost
class MPLS-AF22
  set qos-group mpls experimental topmost
class MPLS-AF31
  set qos-group mpls experimental topmost
class MPLS-AF32
  set qos-group mpls experimental topmost
!
policy-map qos-group-out
class qos-group-AF11
  bandwidth percent 5
  random-detect
  set precedence qos-group
class qos-group-AF12
  bandwidth percent 10
  random-detect
  set precedence qos-group
class qos-group-AF21
  bandwidth percent 10
  random-detect
  set precedence qos-group
class qos-group-AF22
  bandwidth percent 15
  random-detect
  set precedence qos-group
class qos-group-AF31
  bandwidth percent 20
  random-detect
  set precedence qos-group
class qos-group-AF32
  bandwidth percent 30
  random-detect
  set precedence qos-group
!
interface Ethernet0/0
  ip vrf forwarding v1
  ip address 10.0.1.1 255.255.255.0
  max-reserved-bandwidth 90
  service-policy output qos-group-out
!
interface Ethernet1/0
  ip address 192.168.2.2 255.255.255.0
  service-policy input qos-group-in
  tag-switching ip
!

```

## Modo da tubulação

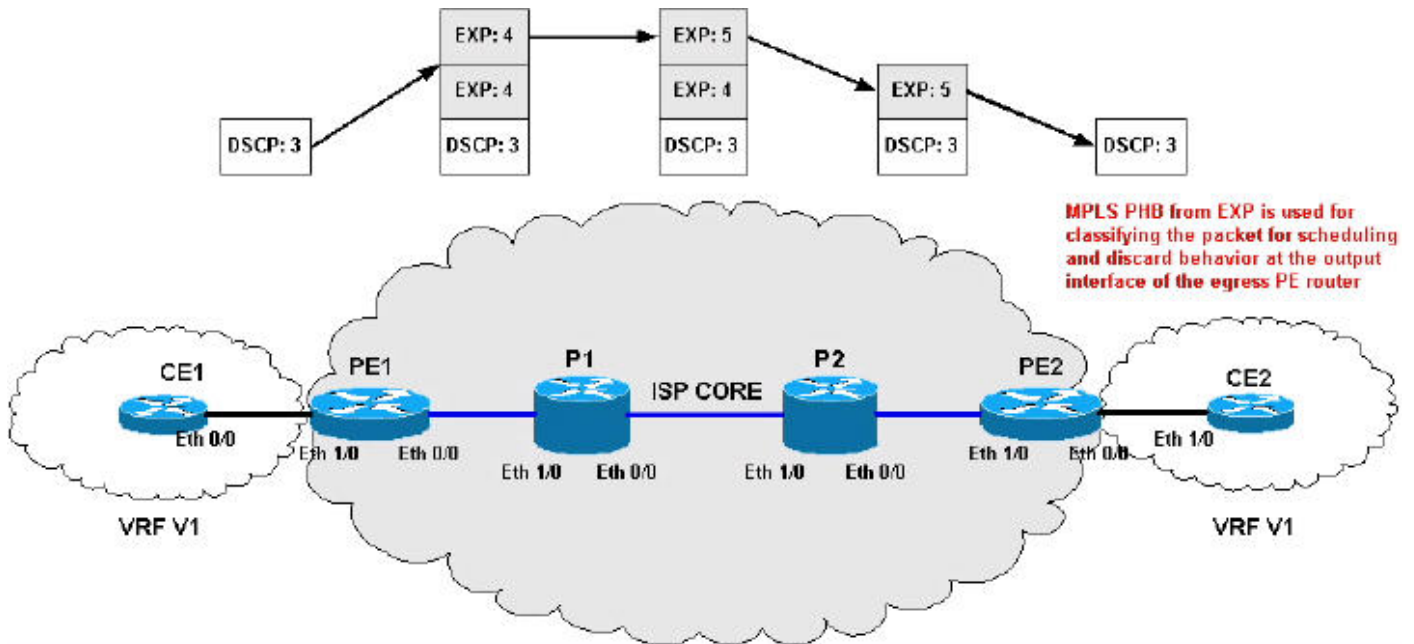
O modo da tubulação do Tunelamento do DiffServ usa duas camadas de QoS:

1. Um QoS subjacente para os dados, que permanecem inalterados ao atravessar o núcleo.
2. Um por-núcleo QoS, que seja separado daquele dos pacotes IP subjacentes. Este por-núcleo QoS PHB permanece transparente aos utilizadores finais.

Quando um pacote alcança a borda do núcleo MPLS, o roteador de PE da saída (PE2) classifica os pacotes IP recentemente expostos para o Enfileiramento de partida baseado no MPLS PHB



dos bit EXP da etiqueta recentemente removida.



Configurações de modo da tubulação:

```

PE1

!--- On input, the IP Precedence is copied to the MPLS
EXP !--- on label imposition and when leaving the
router, the !--- MPLS EXP bits are used to classify the
traffic into three !--- traffic classes—gold, silver,
and bronze (the topmost). class-map match-all IP-AF22
match ip precedence 0 1 class-map match-all IP-AF31
match ip precedence 2 3 class-map match-all IP-AF32
match ip precedence 4 5 ! class-map match-all bronze
match mpls experimental topmost 2 3 class-map match-all
silver match mpls experimental topmost 4 class-map
match-all gold match mpls experimental topmost 5 !
policy-map set-MPLS-PHB
  class IP-AF22
    police 12000 conform-action set-mpls-exp-imposition-
transmit
      3 exceed-action set-mpls-exp-imposition-
transmit 2
  class IP-AF31
    police 12000 conform-action set-mpls-exp-imposition-
transmit
      4 exceed-action set-mpls-exp-imposition-
transmit 3
  class IP-AF32
    police 12000 conform-action set-mpls-exp-imposition-
transmit
      5 exceed-action set-mpls-exp-imposition-
transmit 4
  !
policy-map output-qos
  class gold
    bandwidth 40
    random-detect
  class silver
    bandwidth 30
  
```

```

    random-detect
class bronze
    bandwidth 20
    random-detect
!
interface Ethernet0/0
    ip address 192.168.0.1 255.255.255.0
    max-reserved-bandwidth 90
    service-policy output output-qos
    tag-switching ip
!
interface Ethernet1/0
    ip vrf forwarding v1
    ip address 10.0.0.2 255.255.255.0
    service-policy input set-MPLS-PHB
!

```

## P1

```

!--- This configuration swaps the top label from 4 to 5,
!--- which does not need to occur to follow the previous
configuration. class-map match-all mpls-in match mpls
experimental topmost 4 ! policy-map mpls-in
    class mpls-in
        set mpls experimental topmost 5
!
interface Ethernet0/0
    ip address 192.168.1.1 255.255.255.0
    tag-switching ip
!
interface Ethernet1/0
    ip address 192.168.0.2 255.255.255.0
    service-policy input mpls-in
    tag-switching ip
!

```

## P2

```

!--- Remember to copy down the MPLS EXP value from !---
the newly exposed label after the PHP. class-map match-
all MPLS-AF11 match mpls experimental topmost 0 class-
map match-all MPLS-AF12 match mpls experimental topmost
1 class-map match-all MPLS-AF21 match mpls experimental
topmost 2 class-map match-all MPLS-AF22 match mpls
experimental topmost 3 class-map match-all MPLS-AF31
match mpls experimental topmost 4 class-map match-all
MPLS-AF32 match mpls experimental topmost 5 ! class-map
match-all gold match qos-group 5 class-map match-all
silver match qos-group 4 class-map match-all bronze
match qos-group 3 match qos-group 2 ! policy-map qos-
group-in
    class MPLS-AF11
        set qos-group mpls experimental topmost
    class MPLS-AF12
        set qos-group mpls experimental topmost
    class MPLS-AF21
        set qos-group mpls experimental topmost
    class MPLS-AF22
        set qos-group mpls experimental topmost
    class MPLS-AF31
        set qos-group mpls experimental topmost

```

```

class MPLS-AF32
  set qos-group mpls experimental topmost
!
policy-map qos-group-out
  class gold
    bandwidth 40
    random-detect
    set mpls experimental topmost qos-group
  class silver
    bandwidth 30
    random-detect
    set mpls experimental topmost qos-group
  class bronze
    bandwidth 20
    random-detect
    set mpls experimental topmost qos-group
!
interface Ethernet0/0
  ip address 192.168.2.1 255.255.255.0
  max-reserved-bandwidth 90
  service-policy output qos-group-out
  tag-switching ip
!
interface Ethernet1/0
  ip address 192.168.1.2 255.255.255.0
  service-policy input qos-group-in
  tag-switching ip
!

```

## PE2

*!--- Remember to queue the newly exposed IP packet based in !--- the MPLS EXP bits of the label we just removed. Use !--- qos-groups to keep track of this value.*

```

class-map match-all MPLS-AF11
  match mpls experimental topmost 0
class-map match-all MPLS-AF12
  match mpls experimental topmost 1
class-map match-all MPLS-AF21
  match mpls experimental topmost 2
class-map match-all MPLS-AF22
  match mpls experimental topmost 3
class-map match-all MPLS-AF31
  match mpls experimental topmost 4
class-map match-all MPLS-AF32
  match mpls experimental topmost 5
!
class-map match-all gold
  match qos-group 5
class-map match-all silver
  match qos-group 4
class-map match-all bronze
  match qos-group 3
  match qos-group 2
!
policy-map qos-group-in
  class MPLS-AF11
    set qos-group mpls experimental topmost
    set discard-class 0
  class MPLS-AF12
    set qos-group mpls experimental topmost

```

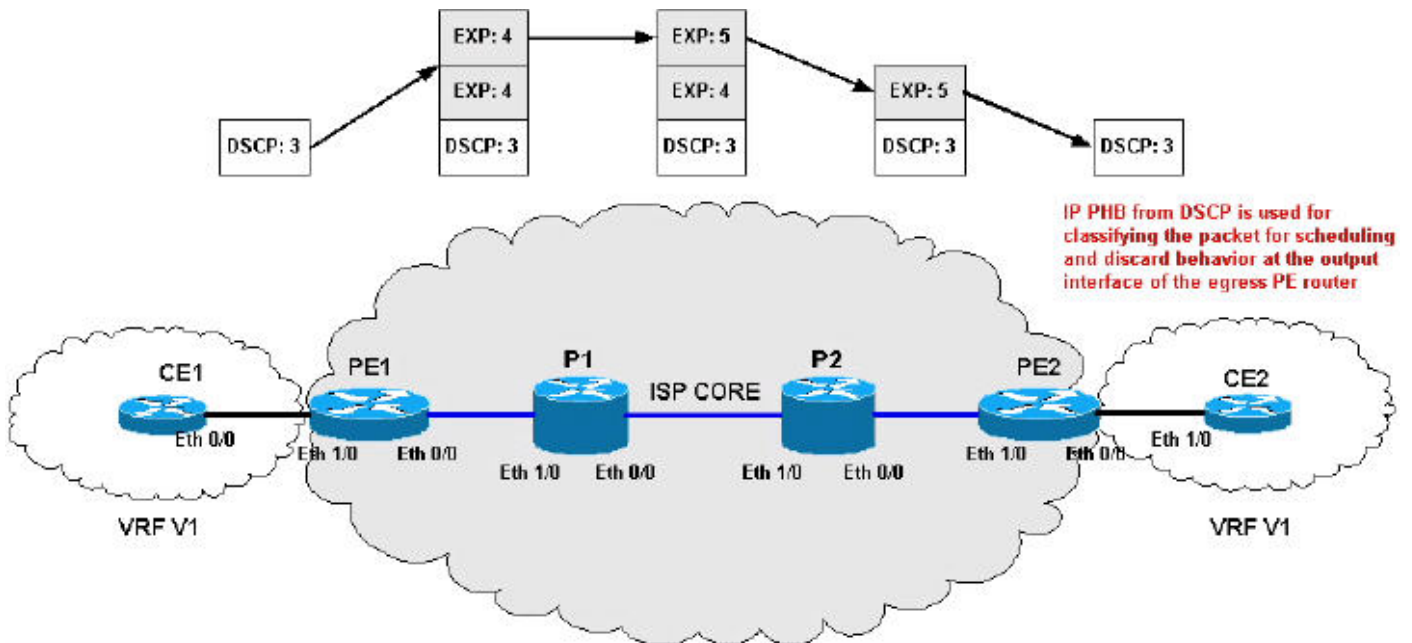
```

    set discard-class 1
class MPLS-AF21
    set qos-group mpls experimental topmost
    set discard-class 2
class MPLS-AF22
    set qos-group mpls experimental topmost
    set discard-class 3
class MPLS-AF31
    set qos-group mpls experimental topmost
    set discard-class 4
class MPLS-AF32
    set qos-group mpls experimental topmost
    set discard-class 5
!
policy-map qos-group-out
class gold
    bandwidth 40
    random-detect discard-class-based
class silver
    bandwidth 30
    random-detect discard-class-based
class bronze
    bandwidth 20
    random-detect discard-class-based
!
interface Ethernet0/0
    ip vrf forwarding v1
    ip address 10.0.1.1 255.255.255.0
    max-reserved-bandwidth 90
    service-policy output qos-group-out
!
interface Ethernet1/0
    ip address 192.168.2.2 255.255.255.0
    service-policy input qos-group-in
    tag-switching ip
!

```

## Modo da Curto-tubulação

O modo da Curto-tubulação do Tunelamento do DiffServ usa as mesmas regras e técnicas através do núcleo. A diferença está no roteador de PE da saída (PE2) — você classifica os pacotes IP recentemente expostos para o Enfileiramento de partida baseado no IP PHB do valor DSCP deste pacote IP.



configurações de modo da Curto-tubulação:

```

PE1

!--- On input, the IP Precedent is copied to the MPLS
EXP !--- on label imposition. When leaving the router,
the !--- MPLS EXP bits are used to classify the traffic
into three !--- traffic classes—gold, silver and bronze
(the topmost). class-map match-all IP-AF22 match ip
precedence 0 1 class-map match-all IP-AF31 match ip
precedence 2 3 class-map match-all IP-AF32 match ip
precedence 4 5 ! class-map match-all bronze match mpls
experimental topmost 2 3 class-map match-all silver
match mpls experimental topmost 4 class-map match-all
gold match mpls experimental topmost 5 ! policy-map set-
MPLS-PHB
  class IP-AF22
    police 12000 conform-action set-mpls-exp-imposition-
transmit
      3 exceed-action set-mpls-exp-imposition-
transmit 2
  class IP-AF31
    police 12000 conform-action set-mpls-exp-imposition-
transmit
      4 exceed-action set-mpls-exp-imposition-
transmit 3
  class IP-AF32
    police 12000 conform-action set-mpls-exp-imposition-
transmit
      5 exceed-action set-mpls-exp-imposition-
transmit 4
!
policy-map output-qos
  class gold
    bandwidth 40
    random-detect
  class silver
    bandwidth 30
    random-detect
  class bronze

```

```

bandwidth 20
random-detect
!
interface Ethernet0/0
ip address 192.168.0.1 255.255.255.0
max-reserved-bandwidth 90
service-policy output output-qos
tag-switching ip
!
interface Ethernet1/0
ip vrf forwarding v1
ip address 10.0.0.2 255.255.255.0
service-policy input set-MPLS-PHB
!

```

## P1

```

!--- This configuration swaps the top label from 4 to 5,
!--- which does not need to occur to follow the previous
configuration. class-map match-all mpls-in match mpls
experimental topmost 4 ! policy-map mpls-in
class mpls-in
set mpls experimental topmost 5
!
interface Ethernet0/0
ip address 192.168.1.1 255.255.255.0
tag-switching ip
!
interface Ethernet1/0
ip address 192.168.0.2 255.255.255.0
service-policy input mpls-in
tag-switching ip
!

```

## P2

```

!--- Remember to copy down the MPLS EXP value from !---
the newly exposed label after the PHP. class-map match-
all MPLS-AF11 match mpls experimental topmost 0 class-
map match-all MPLS-AF12 match mpls experimental topmost
1 class-map match-all MPLS-AF21 match mpls experimental
topmost 2 class-map match-all MPLS-AF22 match mpls
experimental topmost 3 class-map match-all MPLS-AF31
match mpls experimental topmost 4 class-map match-all
MPLS-AF32 match mpls experimental topmost 5 ! class-map
match-all gold match qos-group 5 class-map match-all
silver match qos-group 4 class-map match-all bronze
match qos-group 3 match qos-group 2 ! policy-map qos-
group-in
class MPLS-AF11
set qos-group mpls experimental topmost
class MPLS-AF12
set qos-group mpls experimental topmost
class MPLS-AF21
set qos-group mpls experimental topmost
class MPLS-AF22
set qos-group mpls experimental topmost
class MPLS-AF31
set qos-group mpls experimental topmost
class MPLS-AF32
set qos-group mpls experimental topmost

```

```

!
policy-map qos-group-out
  class gold
    bandwidth 40
    random-detect
    set mpls experimental topmost qos-group
  class silver
    bandwidth 30
    random-detect
    set mpls experimental topmost qos-group
  class bronze
    bandwidth 20
    random-detect
    set mpls experimental topmost qos-group
!
interface Ethernet0/0
  ip address 192.168.2.1 255.255.255.0
  max-reserved-bandwidth 90
  service-policy output qos-group-out
  tag-switching ip
!
interface Ethernet1/0
  ip address 192.168.1.2 255.255.255.0
  service-policy input qos-group-in
  tag-switching ip
!

```

## PE2

```

!--- Remember to queue the newly exposed IP packet !---
based on the value of the IP Precedent. class-map match-
all gold match precedence 4 5 class-map match-all silver
match precedence 2 3 class-map match-all bronze match
precedence 0 1 ! policy-map qos-group-out
  class gold
    bandwidth 40
    random-detect prec-based
  class silver
    bandwidth 30
    random-detect prec-based
  class bronze
    bandwidth 20
    random-detect prec-based
!
interface Ethernet0/0
  ip vrf forwarding v1
  ip address 10.0.1.1 255.255.255.0
  max-reserved-bandwidth 90
  service-policy output qos-group-out
!
interface Ethernet1/0
  ip address 192.168.2.2 255.255.255.0
  tag-switching ip
!

```

## Sumário do modo de túnel

Esta tabela resume as várias ações que se aplicaram ao IP ou aos pacotes rotulados em várias fases na rede:

Modo da escavação de um túnel	IP -> etiqueta	Etiqueta -> etiqueta	Etiqueta -> IP
Uniforme	Copie o Prec/DiffServ IP em MPLS EXP (pode ser mudado pelo SP igualmente)	O MPLS EXP pode ser mudado pelo SP	MPLS EXP copiado ao Prec/DiffServ IP
Tubulação	MPLS EXP ajustado pela política de QoS SP		IP original Prec/DiffServ preservado (Enfileiramento da saída baseado em MPLS EXP)
Curto-tubulação			IP original Prec/DiffServ preservado (Enfileiramento da saída baseado no Prec/DiffServ IP)

## [Informações Relacionadas](#)

- [Cisco IOS Software - MPLS](#)
- [Q &A - Qualidade de Serviço para redes Multiprotocol Label Switching](#)
- [Referência de comandos dos Serviços de comutação Cisco IOS, Versão 12.2](#)
- [Manual de configuração das soluções da Qualidade de serviço Cisco IOS, Versão 12.2](#)
- [Referência do comando solutions da Qualidade de serviço Cisco IOS, Versão 12.2](#)
- [Página de suporte de MPLS](#)
- [página de suporte de QoS](#)
- [Suporte Técnico - Cisco Systems](#)