

# MPLS ネットワーク用の DiffServ トンネリングモード

## 目次

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[背景説明](#)

[デフォルト動作](#)

[Cisco IOS ソフトウェア 12.2\(13\)T よりも前のコマンドの使用方法和動作](#)

[Cisco IOS ソフトウェア 12.2\(13\)T 以降のコマンドの使用方法和動作](#)

[DiffServ トンネリング モード](#)

[均一モード](#)

[パイプ モード](#)

[ショートパイプ モード](#)

[トンネル モードの概要](#)

[関連情報](#)

## [はじめに](#)

このドキュメントでは、マルチプロトコル ラベル スイッチング ( MPLS ) ベースのネットワーク環境で使用可能な差別化サービス ( DiffServ ) トンネリング モードの実装について説明します。

## [前提条件](#)

### [要件](#)

このドキュメントの読者は次のトピックについて理解している必要があります。

- MPLS およびバーチャル プライベート ネットワーク ( VPN ) 用 MPLS
- IP プレシデンス、タイプ オブ サービス ( ToS )、および DiffServ に関する概念
- モジュラ QoS コマンドライン インターフェイス ( MQC ) を使用した Quality of Service ( QoS ) パケット マーキングおよび分類

### [使用するコンポーネント](#)

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づくものです。

- 新しい QoS 機能に関しては Cisco IOS ソフトウェア<sup>®</sup> リリース 12.2(13)T、元の QoS 機能に関してはリリース 12.1(5)T。
- MPLS プロバイダー コア ( P ) ルータ/プロバイダー エッジ ( PE ) ルータの機能をサポートする 3600 シリーズ以降のシスコ ルータ ( Cisco 3660 や 7206 など ) 。

本書の情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。このドキュメントで使用するすべてのデバイスは、初期 ( デフォルト ) 設定の状態から起動しています。稼働中のネットワークで作業を行う場合、コマンドの影響について十分に理解したうえで作業してください。

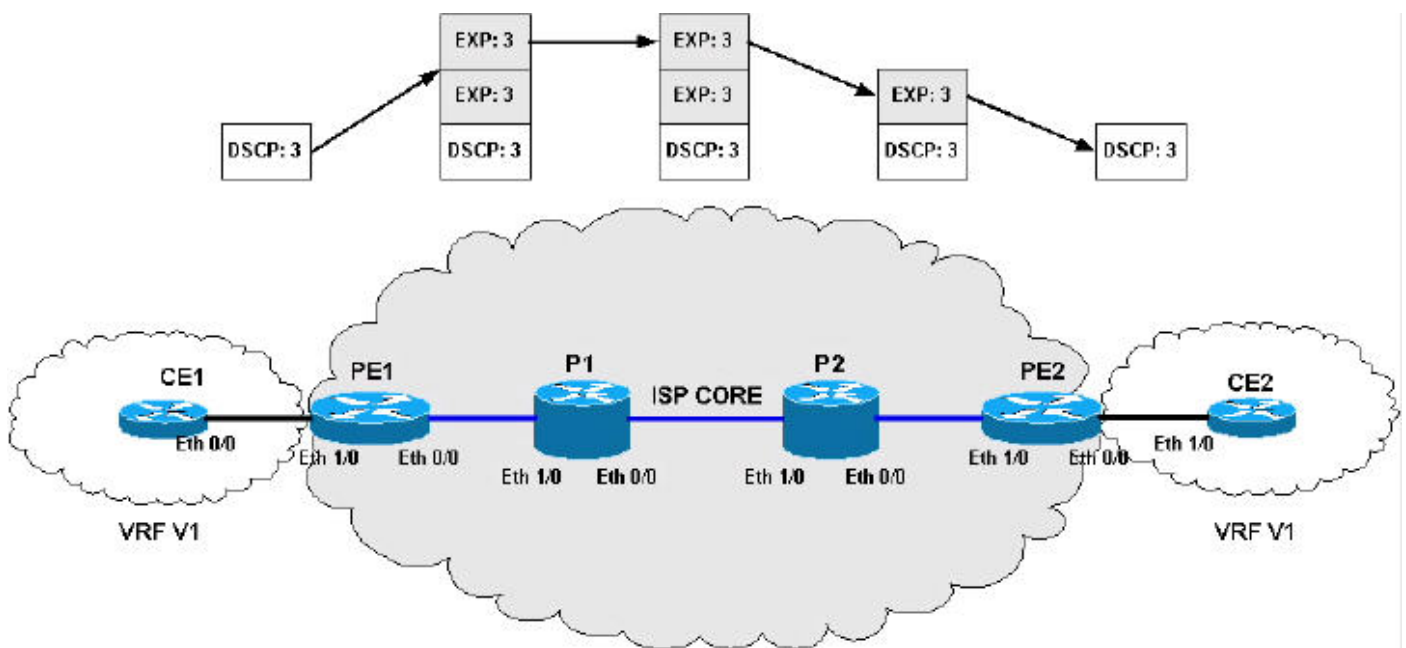
## 表記法

ドキュメント表記の詳細は、『[シスコテクニカルティップスの表記法](#)』を参照してください。

## 背景説明

### デフォルト動作

次の図は、パケットが MPLS コアでカスタマー エッジ ( CE ) ルータ間を移動する際の DiffServ コードポイント ( DSCP ) ビットおよび MPLS Experimental ( EXP ) ビットのデフォルト動作を示しています。



この項では、「デフォルト動作」の図に示した動作について説明していきます。

ラベルのインポジション ( IP -> ラベル ) :

- 着信 IP パケットの IP 優先順位が、プッシュされたすべてのラベルの MPLS EXP ビットにコピーされます。
- DSCP ビットの最初の 3 ビットが、プッシュされたすべてのラベルの MPLS EXP ビットにコピーされます。
- この手法は ToS Reflection ( TOS の反映 ) とも呼ばれています。

MPLS 転送 ( ラベル -> ラベル ) :

- 転送またはインポジション中にスワップ/プッシュされた新しいラベルに EXP がコピーされます。
- ラベル インポジションでは、現在のラベル スタックに追加される新しいラベルの値によって元のラベルが変更されることはありません。
- ラベルのディスポジション ( disposition ) では、新たに表れるようになったラベルの EXP ビットに、取り外したラベルの EXP ビットはコピーされません。

ラベルのディスポジション ( ラベル -> IP ) :

- ラベル ディスポジションで、新たにエクスポートされた IP パケットの IP precedence または DSCP フィールドに EXP ビットがコピーされることはありません。

## Cisco IOS ソフトウェア 12.2(13)T よりも前のコマンドの使用方法和動作

IOS リリース 12.2(13)T よりも前のリリースでは、MPLS EXP ビットを変更するには、[set mpls experimental](#) コマンドが唯一の方法でした。

## Cisco IOS ソフトウェア 12.2(13)T 以降のコマンドの使用方法和動作

IOS リリース 12.2(13)T 以降のリリースでは、次のオプションを使用できるように [set mpls experimental](#) コマンドが変更されています。

- [set mpls experimental topmost {mpls-exp-value / qos-group \[table table-map-name\]}](#)
- [set mpls experimental imposition {mpls-exp-value / qos-group \[table table-map-name\]}](#)

注: 新しい [set mpls experimental topmost](#) コマンドは、以前の [set mpls imposition](#) コマンドと同等です。

この 2 つのコマンドを新しいコマンド スイッチと組み合わせて使用すると、ラベルのプッシュ、スワップ、およびポップ処理時の MPLS EXP ビットの動作をより適切に制御できます。この 2 つのコマンドでは DiffServ トンネリング モードを使用できます。

## DiffServ トンネリング モード

Diffserv トンネリング モードで導入される新しい Per-Hop Behavior ( PHB ) により、プロバイダー ネットワークでの差別化 QoS が可能になります。トンネリング モードはネットワークのエッジで定義されるもので、通常は PE ラベル スイッチ ルータ ( LSR ) の入力と出力の両方で定義されます。P ルータでの変更が必要な場合がありますが、その際は、Penultimate-Hop-Popping ( PHP ) によりパケットから最上位ラベルが削除されたときの動作も考慮する必要があります。ポップされる最上位ラベルの MPLS EXP 値を新たにエクスポートされたラベルにコピーする必要が生じる可能性があります。これは、必ずしもすべてのトンネリング モードに当てはまるわけではありません。

場合によっては ( たとえば、VPN 以外の一般的な MPLS ネットワーク )、ラベルが 1 つのみのパケットを受信すると、最後の P ルータの PHP 処理でプレーン IP パケットがエクスポートされることがあります。出力 LSR ( PE ) でこの IP パケットを受信しても、ラベルが付いていないため、MPLS EXP ビットに基づいてパケットを分類することができません。そのような場合には、[explicit-null](#) ラベルをアドバタイズするように、出口 PE ルータを設定する必要があります。P ルータで PHP 処理が実行されると、値が 0 のラベルが送信されます。この特殊なラベルを使用して、通常のラベル付きパケットとして EXP ビットをマーキングすることで、出力 PE ルータでの適切な分類が可能になります。

Diffserv 仕様の MPLS ネットワーク サポートでは、次のトンネリング モードが定義されています。

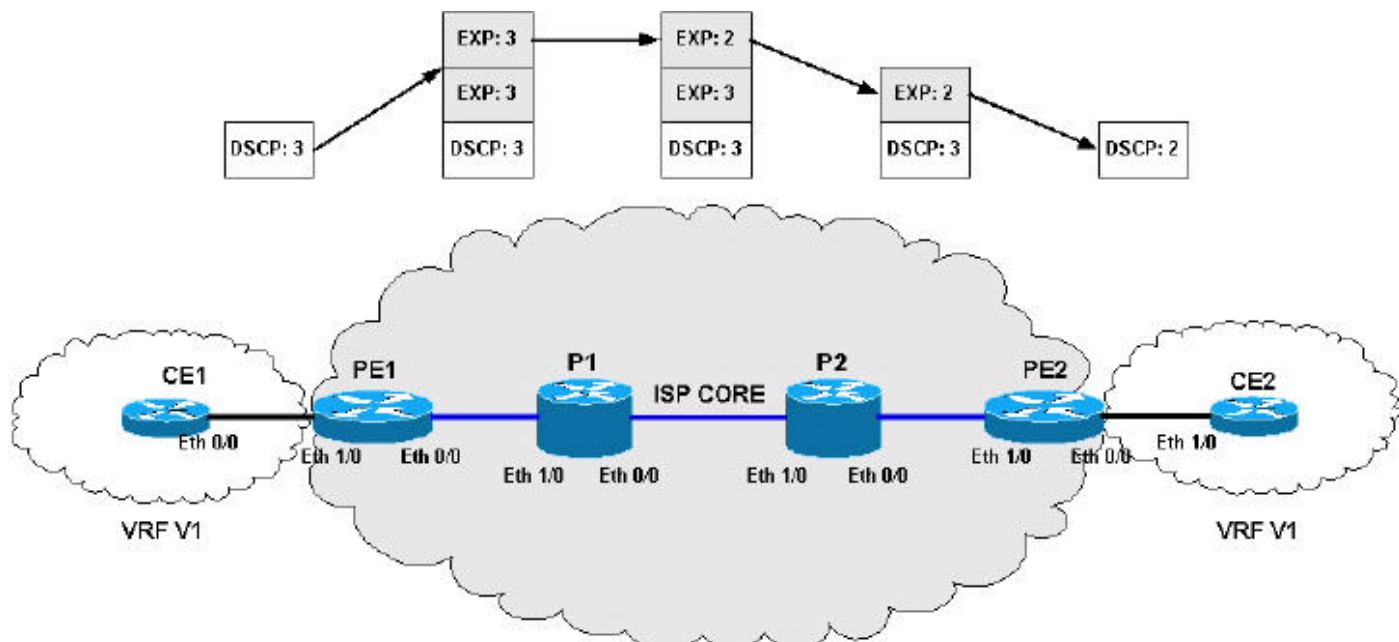
- [均一](#)
- [パイプ](#)
- [ショートパイプ](#)

次の項では各トンネリング モードを個別に検証し、各トンネリング モードの設定方法を示す例を紹介しします。例には、IP プレシデンスの MPLS EXP ビットへの完全なマッピングも含まれています。カスタマーごとに異なる複数の QoS パラメータとトンネリング モードを設定することもできます。

注: 設定例は MPLS VPN に固有のものではなく、一般的な MPLS ネットワークと Carrier Supporting Carrier ( CSC ) ネットワークにも適用できます。また、多数のさまざまなパラメータとトンネリング モードを使用できるため、ネットワークごとに異なる可能性があります。

## 均一モード

DiffServ トンネリングの均一モードには、エンドツーエンドに到達する QoS のレイヤが 1 つだけ存在します。入力 PE ルータ ( PE1 ) は、インポートされたラベルの MPLS EXP ビットに着信 IP パケットの DSCP をコピーします。EXP ビットはコアを通過するときに、中間 P ルータによって変更される場合とされない場合があります。この例では、P ルータ P1 は最上位ラベルの EXP ビットを変更します。出力 P ルータ ( P2 ) では、PHP ( Penultimate-Hop-Pop ) 後に新たにエクスポートされたラベルの EXP ビットに EXP ビットをコピーします。最後に出力 PE ルータ ( PE2 ) で、新たにエクスポートされた IP パケットの DSCP ビットに EXP ビットをコピーします。



均一モードの設定 :

```
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-gos
  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
!
```

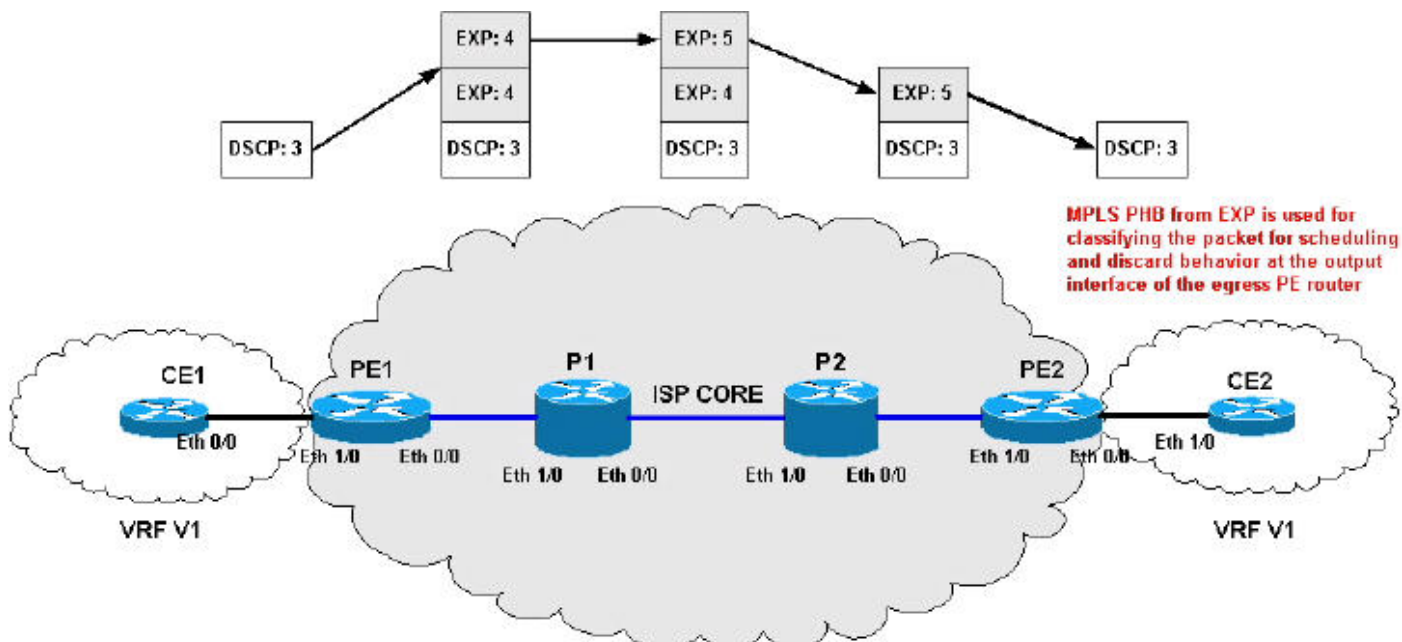
## パイプモード

Diffserv トンネリングのパイプモードは、次の2つのQoSレイヤを使用します。

1. データの元のQoS。コアを通過してもそのまま変更されません。
2. コアごとのQoS。元のIPパケットのQoSとは別のQoSです。このコアごとのQoSのPHBは、エンドユーザに対して透過的なままです。

パケットがMPLSコアのエッジに到達すると、出力PEルータ(PE2)は、直前に削除されたレベルのEXPビットから取得したMPLS PHBに基づいて、新たにエクスポートされたIPパケットを発信キュー用に分類します。





パイプ モードの設定 :

```

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

```

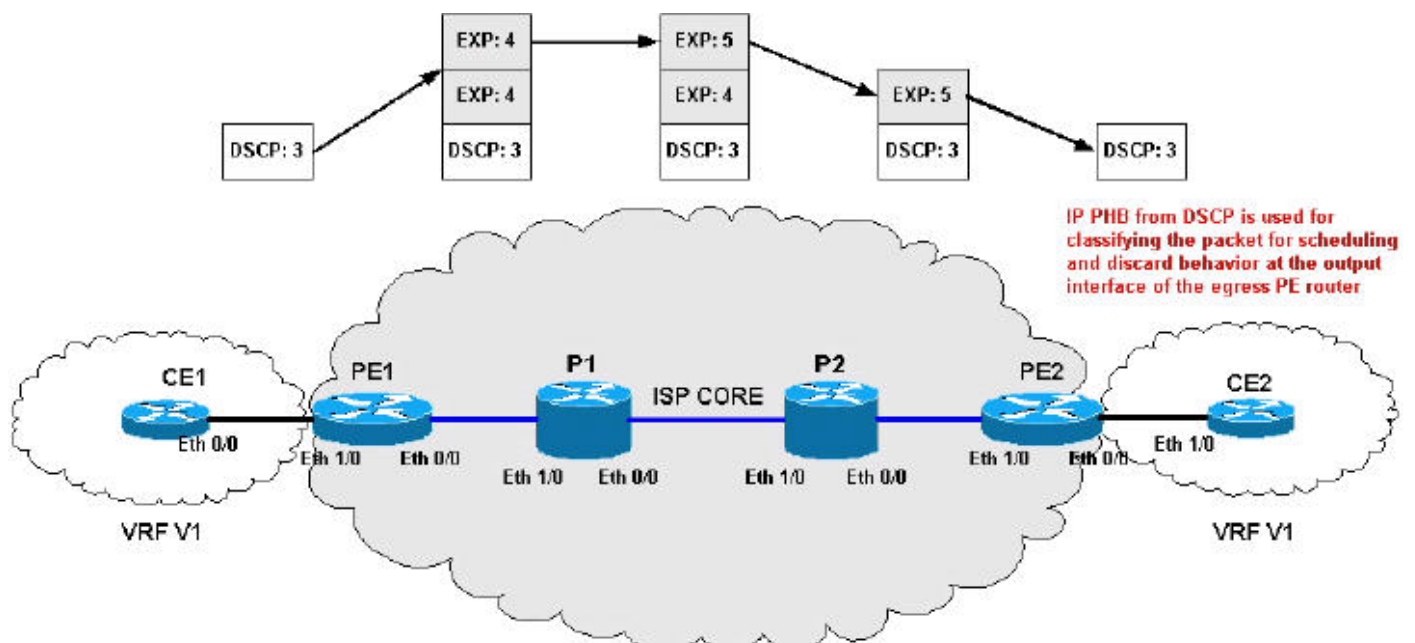
```

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
!

```

## ショートパイプ モード

Diffserv トンネリングのショートパイプ モードは、コア全体で同じルールと技術を使用します。異なるのは、出力 PE ルータ ( PE2 ) で、新たにエクスポートされた IP パケットを発信キュー用に分類する際に、この IP パケットの DSCP 値から取得した IP PHB が使用される点です。



## シヨートパイプ モードの設定 :

### 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
!

```

## トンネル モードの概要

次の表に、ネットワークの各段階で IP パケットまたはラベル付きパケットに対して行われるさまざまな処理をまとめます。

トンネリングモード	IP -> ラベル	ラベル -> ラベル	ラベル -> IP
ユニフォーム	IP Prec/DiffServ を MPLS EXP にコピーする ( SP によって変更される場合もある )	MPLS EXP が SP によって変更される場合がある	MPLS EXP が IP Prec/DiffServ にコピーされる
Pipe	MPLS EXP が SP の QoS ポリシーによって設定される		元の IP Prec/DiffServ が維持される ( MPLS EXP に基づいた出力キューイング )

ショート パイプ			元の IP Prec/DiffServ が維持される ( IP Prec/DiffServ に基づいた出 力キューイン グ )
-------------	--	--	--

## 関連情報

- [Cisco IOS ソフトウェア : MPLS](#)
- [Q&A : マルチプロトコル ラベル スイッチング ネットワークの Quality of Service](#)
- [Cisco IOS スイッチング サービス リリース 12.2 コマンド リファレンス](#)
- [Cisco IOS Quality of Service ソリューション リリース 12.2 コンフィギュレーション ガイド](#)
- [Cisco IOS Quality of Service ソリューション リリース 12.2 コマンド リファレンス](#)
- [MPLS に関するサポートページ](#)
- [QoS に関するサポート ページ](#)
- [テクニカルサポート - Cisco Systems](#)