Configuring MPLS Basic Traffic Engineering Using IS-IS

Please provide your feedback on this document.

Introduction

This sample configuration shows how to implement traffic engineering (TE) on top of a existing Multiprotocol Label Switching (MPLS) network using Frame Relay and Intermediate System-to-Intermediate System (IS-IS). Our example implements two dynamic tunnels (automatically set up by the ingress Label Switch Routers (LSR)) and two tunnels that use explicit paths.

TE is a generic name corresponding to the use of different technologies to optimize the utilization of a given backbone capacity and topology.

MPLS TE provides a way to integrate TE capabilities (such as those used on Layer 2 protocols like ATM) into Layer 3 protocols (IP). MPLS TE uses an extension to existing protocols (Resource Reservation Protocol (RSVP), IS-IS, Open Shortest Path First (OSPF)) to calculate and establish unidirectional tunnels that are set according to the network constraint. Traffic flows are mapped on the different tunnels depending on their destination.

Functional Components

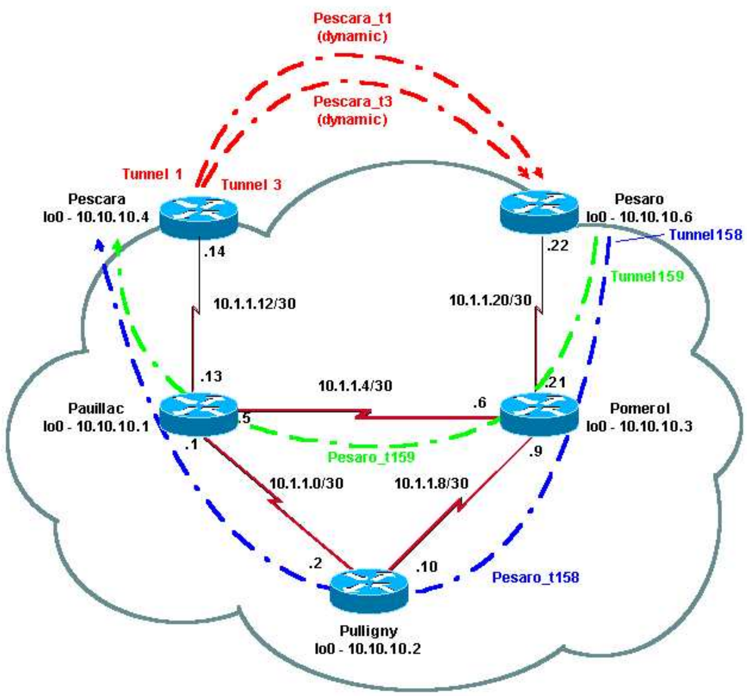
Component	Description
IP tunnel interfaces	Layer 2: an MPLS tunnel interface is the head of a Label Switched Path (LSP). It is configured with a set of resource requirements, such as bandwidth and priority. Layer 3: the LSP tunnel interface is the head-end of a unidirectional virtual link to the tunnel destination.
RSVP with TE extension	RSVP is used to establish and maintain LSP tunnels based on the calculated path using PATH and RESV messages. The RSVP protocol specification has been extended so that the RESV messages also distribute label information.
Link-state IGP (IS-IS or OSPF with TE extension)	Used to flood topology and resource information from the link management module. IS-IS uses new Type-Length-Values (TLVs); OSPF uses type 10 Link State Advertisements (also called Opaque LSAs).
MPLS TE path calculation module	Operates at the LSP head only and determines a path using information from the link-state database.
MPLS TE link management module	At each LSP hop, this module performs link call admission on the RSVP signaling messages, and bookkeeping of topology and resource information to be flooded by OSPF or IS-IS.
Label switching forwarding	Basic MPLS forwarding mechanism based on labels.

Hardware and Software Versions

This configuration was developed and tested using the software and hardware versions below.

- Cisco IOS® Software Releases 12.0(11)S and 12.1(3a)T
- Cisco 3600 routers

Network Diagram



Configurations

Quick Configuration Guide

The following steps can be used to perform a quick configuration. For more detailed information, see [MPLS Traffic Engineering and Enhancements](#).

1. Set up your network with the usual configuration (in this case, we used Frame Relay). Note: It is mandatory to set up a loopback interface with a IP mask of 32 bits. This address will be used for the set up of the MPLS network and TE by the routing protocol. This loopback address must be reachable via the global routing table.
2. Set up a routing protocol for the MPLS network. It must be a link-state protocol (IS-IS or OSPF). In the routing protocol configuration mode, enter:
   - For IS-IS:

     ```
     metric-style wide (or metric-style both)
     mpls traffic-eng router-id LoopbackN
     mpls traffic-eng [level-1 | level-2 |]
     ```

   - For OSPF:

     ```
     mpls traffic-eng area X
     mpls traffic-eng router-id LoopbackN (must have a 255.255.255.255 mask)
     ```

3. Enable MPLS TE. Enter **ip cef** (or **ip cef distributed** if available in order to enhance performance) in the general configuration mode. Enable MPLS (**tag-switching ip**) on each concerned interface. Enter **mpls traffic-engineering tunnel** to enable MPLS TE.

4. Enable RSVP by entering **ip rsvp bandwidth XXX** on each concerned interface.

5. Set up tunnels to be used for TE. There are many options that can be configured for MPLS TE Tunnel, but the **tunnel mode mpls traffic-eng** command is mandatory. The **tunnel mpls traffic-eng autoroute announce** command announces the presence of the tunnel by the routing protocol.

Note: Don’t forget to use **ip unnumbered loopbackN** for the IP address of the tunnel interfaces.

This sample configuration shows two dynamic tunnels with different bandwidth (and priorities) going from the Pescara router to the Pesaro router, and two tunnels using an explicit path going from Pesaro to Pescara.

Configuration File

Only the relevant parts of the configuration files are included. The commands used to enable MPLS are underlined, while the commands specific to TE (including RSVP) are in bold.

Pesaro

```
Current configuration:
 !
 version 12.1
 !
 hostname Pesaro
 !
 ip cef
 mpls traffic-eng tunnels
 !
 interface Loopback0
  ip address 10.10.10.6 255.255.255.255
  ip router isis
 !
 interface Tunnel158
  ip unnumbered Loopback0
  tunnel destination 10.10.10.4
  tunnel mode mpls traffic-eng
  tunnel mpls traffic-eng autoroute announce
  tunnel mpls traffic-eng priority 2 2
  tunnel mpls traffic-eng bandwidth 158
  tunnel mpls traffic-eng path-option 1 explicit name low
 !
 interface Tunnel159
  ip unnumbered Loopback0
  tunnel destination 10.10.10.4
  tunnel mode mpls traffic-eng
  tunnel mpls traffic-eng autoroute announce
  tunnel mpls traffic-eng priority 4 4
  tunnel mpls traffic-eng bandwidth 159
  tunnel mpls traffic-eng path-option 1 explicit name straight
 !
 interface Serial0/0
  no ip address
  encapsulation frame-relay
 !
 interface Serial0/0.1 point-to-point
  bandwidth 512
  ip address 10.1.1.22 255.255.255.252
  ip router isis
  tag-switching ip
  mpls traffic-eng tunnels
```

```
 frame-relay interface-dlci 603
ip rsvp bandwidth 512 512
!
router isis
 net 49.0001.0000.0000.0006.00
 is-type level-1
 metric-style wide
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng level-1
!
!
ip classless
!
ip explicit-path name low enable
 next-address 10.1.1.21
 next-address 10.1.1.10
 next-address 10.1.1.1
 next-address 10.1.1.14
!
ip explicit-path name straight enable
 next-address 10.1.1.21
 next-address 10.1.1.5
 next-address 10.1.1.14
!
end
```

Pescara

```
Current configuration:
!
version 12.0
!
hostname Pescara
!

ip cef
!
mpls traffic-eng tunnels
!
interface Loopback0
 ip address 10.10.10.4 255.255.255.255
 ip router isis
!
interface Tunnel1
 ip unnumbered Loopback0

 tunnel destination 10.10.10.6
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 5 5
 tunnel mpls traffic-eng bandwidth 25
 tunnel mpls traffic-eng path-option 2 dynamic
!
interface Tunnel3
 ip unnumbered Loopback0

 tunnel destination 10.10.10.6
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng bandwidth  69
 tunnel mpls traffic-eng path-option 1 dynamic
!
interface Serial0/1
```

```
 no ip address
 encapsulation frame-relay
!
interface Serial0/1.1 point-to-point
 bandwidth 512
 ip address 10.1.1.14 255.255.255.252

 ip router isis
 mpls traffic-eng tunnels
 tag-switching ip
 frame-relay interface-dlci 401
 ip rsvp bandwidth 512 512
!
router isis
 net 49.0001.0000.0000.0004.00
 is-type level-1
 metric-style wide
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng level-1
!
end
```

Pomerol

```
Current configuration:

version 12.0
!
hostname Pomerol
!
ip cef
!
mpls traffic-eng tunnels
!
interface Loopback0
 ip address 10.10.10.3 255.255.255.255
 ip router isis
!
interface Serial0/1
 no ip address
 encapsulation frame-relay
!
interface Serial0/1.1 point-to-point
 bandwidth 512
 ip address 10.1.1.6 255.255.255.252
 ip router isis
 mpls traffic-eng tunnels
 tag-switching ip
  frame-relay interface-dlci 301
 ip rsvp bandwidth 512 512
!
interface Serial0/1.2 point-to-point
 bandwidth 512
 ip address 10.1.1.9 255.255.255.252
 ip router isis
 mpls traffic-eng tunnels
 tag-switching ip
 frame-relay interface-dlci 302
 ip rsvp bandwidth 512 512
!
interface Serial0/1.3 point-to-point
 bandwidth 512
 ip address 10.1.1.21 255.255.255.252
 ip router isis
```

```
  mpls traffic-eng tunnels
  tag-switching ip
  frame-relay interface-dlci 306
  ip rsvp bandwidth 512 512
 !
 router isis
  net 49.0001.0000.0000.0003.00
  is-type level-1
  metric-style wide
  mpls traffic-eng router-id Loopback0
  mpls traffic-eng level-1
 !
 ip classless
 !
 end
```

Pulligny

```
Current configuration:
 !
 version 12.1
 !
 hostname Pulligny
 !
 ip cef
 !
 mpls traffic-eng tunnels
 !
 interface Loopback0
  ip address 10.10.10.2 255.255.255.255
 !
 interface Serial0/1
  no ip address
  encapsulation frame-relay
 !
 interface Serial0/1.1 point-to-point
  bandwidth 512
  ip address 10.1.1.2 255.255.255.252
  ip router isis
  mpls traffic-eng tunnels
  tag-switching ip
  frame-relay interface-dlci 201
  ip rsvp bandwidth 512 512
 !
 interface Serial0/1.2 point-to-point
  bandwidth 512
  ip address 10.1.1.10 255.255.255.252
  ip router isis
  mpls traffic-eng tunnels
  tag-switching ip
  frame-relay interface-dlci 203
  ip rsvp bandwidth 512 512
 !
 router isis
  passive-interface Loopback0
  net 49.0001.0000.0000.0002.00
  is-type level-1
  metric-style wide
  mpls traffic-eng router-id Loopback0
  mpls traffic-eng level-1
 !
 ip classless
 !
 end
```

Pauillac

```
!
 version 12.1
 !
 hostname pauillac
 !
 ip cef
 mpls traffic-eng tunnels
 !
 interface Loopback0
  ip address 10.10.10.1 255.255.255.255
  ip router isis
 !
 interface Serial0/0
  no ip address
  encapsulation frame-relay
 !
 interface Serial0/0.1 point-to-point
  bandwidth 512
  ip address 10.1.1.1 255.255.255.252
  ip router isis
  mpls traffic-eng tunnels
  tag-switching ip
  frame-relay interface-dlci 102
  ip rsvp bandwidth 512 512
 !
 interface Serial0/0.2 point-to-point
  bandwidth 512
  ip address 10.1.1.5 255.255.255.252
  ip router isis
  mpls traffic-eng tunnels
  tag-switching ip
  frame-relay interface-dlci 103
  ip rsvp bandwidth 512 512
 !
 interface Serial0/0.3 point-to-point
  bandwidth 512
  ip address 10.1.1.13 255.255.255.252
  ip router isis
  mpls traffic-eng tunnels
  tag-switching ip
  frame-relay interface-dlci 104
  ip rsvp bandwidth 512 512
 !
 router isis
  net 49.0001.0000.0000.0001.00
  is-type level-1
  metric-style wide
  mpls traffic-eng router-id Loopback0
  mpls traffic-eng level-1
 !
 ip classless
 !
 end
```

show Commands

The following commands are illustrated below:

- **show mpls traffic-eng tunnels brief**

- **show mpls traffic-eng tunnels name Pesaro_t158**
- **show ip rsvp interface**
- **show mpls traffic-eng topology path destination 10.10.10.6 bandwidth 75**

Other useful commands (not illustrated here) include:

- **show isis mpls traffic-eng advertisements**
- **show tag-switching forwarding-table**
- **show ip cef**
- **show mpls traffic-eng tunnels summary**

Sample show Output

On any LSR, you can use **show mpls traffic-eng tunnels** to check the existence and state of the tunnels. For example, on Pesaro, we see a total of four tunnels, two arriving at Pesaro (shown in red, Pescara_t1 and t3) and two starting from Pesaro (t158 (shown in blue) and t159 (shown in green)):

```
Pesaro#show mpls traffic-eng tunnels brief
Signaling Summary:
    LSP Tunnels Process:            running
    RSVP Process:                   running
    Forwarding:                     enabled
    Periodic reoptimization:        every 3600 seconds, next in 606 seconds
TUNNEL NAME                      DESTINATION      UP IF     DOWN IF   STATE/PROT
Pesaro_t158                      10.10.10.4       -         Se0/0.1   up/up
Pesaro_t159                      10.10.10.4       -         Se0/0.1   up/up
Pescara_t1                       10.10.10.6       Se0/0.1   -         up/up
Pescara_t3                       10.10.10.6       Se0/0.1   -         up/up
Displayed 2 (of 2) heads, 0 (of 0) midpoints,2 (of 2) tails
```

While on a middle router we see the following:

```
Pulligny#show mpls traffic-eng tunnels brief
Signaling Summary:
    LSP Tunnels Process:            running
    RSVP Process:                   running
    Forwarding:                     enabled
    Periodic reoptimization:        every 3600 seconds, next in 406 seconds
TUNNEL NAME                      DESTINATION      UP IF     DOWN IF   STATE/PROT
Pescara_t3                       10.10.10.6       Se0/1.1   Se0/1.2   up/up
Pesaro_t158                      10.10.10.4       Se0/1.2   Se0/1.1   up/up
Displayed 0 (of 0) heads, 2 (of 2) midpoints, 0 (of 0) tails
```

The detailed configuration of any tunnel can be seen using the following:

```
Pesaro#show mpls traffic-eng tunnels name Pesaro_t158

Name: Pesaro_t158                          (Tunnel158) Destination: 10.10.10.4
  Status:
    Admin: up         Oper: up     Path: valid       Signaling: connected

    path option 1, type explicit low (Basis for Setup, path weight 40)

  Config Parameters:
    Bandwidth:  158      kbps  Priority: 2  2   Affinity: 0x0/0xFFFF
    AutoRoute:  enabled   LockDown: disabled
```

```
  InLabel  :  -
  OutLabel : Serial0/0.1, 17
  RSVP Signaling Info:
       Src 10.10.10.6, Dst 10.10.10.4, Tun_Id 158, Tun_Instance 1601
    RSVP Path Info:
      My Address: 10.10.10.6
      Explicit Route: 10.1.1.21 10.1.1.10 10.1.1.1 10.1.1.14
 10.10.10.4
      Record   Route:  NONE
      Tspec: ave rate=158 kbits, burst=8000 bytes, peak rate=158 kbits
    RSVP Resv Info:
      Record   Route:  NONE
      Fspec: ave rate=158 kbits, burst=8000 bytes, peak rate=4294967 kbits
  History:
    Current LSP:
      Uptime: 3 hours, 33 minutes
      Selection: reoptimation
    Prior LSP:
      ID: path option 1 [1600]
      Removal Trigger: configuration changed
```

In this case, the path is explicit and specified in the RSVP message (the field that carries the path is also known as the Explicit Route Object (ERO)). If this path cannot be followed, the MPLS TE engine uses the next path option, which can be another explicit route or a dynamic route.

RSVP specific information is available using standard RSVP commands. In the following output, we see that, on Pulligny, there are two reservations made, one by Pesaro_t158 (158K, shown in blue) and the other by Pescara_t3 (69k, shown in red).

```
Pulligny#show ip rsvp interface
interface     allocated  i/f max  flow max pct UDP  IP   UDP_IP   UDP M/C
Se0/1         0M         0M       0M       0   0    0    0        0
Se0/1.1       158K       512K     512K     30  0    1    0        0
Se0/1.2       69K        512K     512K     13  0    1    0        0
```

If you want to know which the TE path will be used for a particular destination (and a particular bandwidth) without creating a tunnel, you can use the following:

```
Pescara#show mpls traffic-eng topology path destination 10.10.10.6 bandwidth 75
Query Parameters:
  Destination: 10.10.10.6
    Bandwidth: 75
   Priorities: 0 (setup), 0 (hold)
     Affinity: 0x0 (value), 0xFFFFFFFF (mask)
Query Results:
  Min Bandwidth Along Path: 385 (kbps)
  Max Bandwidth Along Path: 512 (kbps)
  Hop  0: 10.1.1.14      : affinity 00000000, bandwidth 512 (kbps)
  Hop  1: 10.1.1.5       : affinity 00000000, bandwidth 385 (kbps)
  Hop  2: 10.1.1.21      : affinity 00000000, bandwidth 512 (kbps)
  Hop  3: 10.10.10.6
```

Finally, if the network does IP TTL propagation (see **mpls ip ttl propagate**), you can do a **traceroute** and see that the path followed is the tunnel and that the tunnel routes according to what has been configured:

```
Pescara#traceroute 10.10.10.6

Type escape sequence to abort.
Tracing the route to 10.10.10.6

  1 10.1.1.13 [MPLS: Label 29 Exp 0] 540 msec 312 msec 448 msec
  2 10.1.1.2 [MPLS: Label 27 Exp 0] 260 msec 276 msec 556 msec
  3 10.1.1.9 [MPLS: Label 29 Exp 0] 228 msec 244 msec 228 msec
```

```
    4 10.1.1.22 112 msec *  104 msec
```

Related Information

- MPLS Support Page

Help us help you.

Please rate this document.

- Excellent
- Good
- Average
- Fair
- Poor

This document solved my problem.

- Yes
- No
- Just browsing

Suggestions for improvement:

(256 character limit)

Optional contact info:

Name:

Email:

Send

Updated: Feb 14, 2003 Document ID: 13737