Applications of MPLS

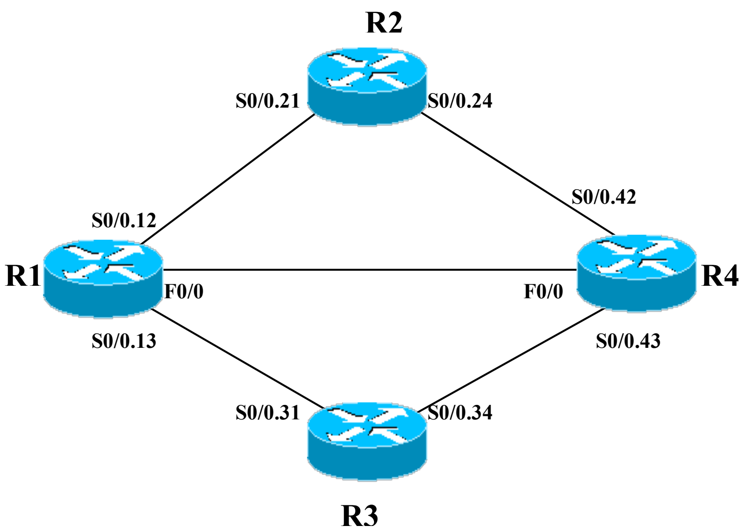
WWW.Net-WorkBooks.com

Narbik Kocharians
CCIE #12410
R&S, Security, SP

Paul Negron
CCIE #14856
SP

MPLS-TE

Lab 1 – Configuring Basic MPLS TE



Router	Interface	IP address
R1	Lo0 S0/0.12 S0/0.13 F0/0	1.1.1.1 /32 10.1.12.1 /24 10.1.13.1 /24 10.1.14.1/ 24
R2	Lo0 S0/0.21 S0/0.23	2.2.2.2 /32 10.1.12.2 /24 10.1.24.2 /24
R3	Lo0 S0/0.31 S0/0.34	3.3.3.3/ 32 10.1.13.3 /24 10.1.34.3 /24
R4	Lo0 F0/0 S0/0.42 S0/0.43	4.4.4.4 /32 10.1.14.4/ 24 10.1.24.4 /24 10.1.34.4 /24

Lab Setup:

The connections between R1 to R2, R1 to R3, R2 to R4, and R3 to R4 are to be setup in a Frame-Relay Point-to-Point manner

Task 1

Configure an OSPF routed core using process "1". Support mpls traffic engineering throughout the core topology.

Configure basic traffic Engineering on ALL Core Devices:

On R1

R1(config)# mpls traffic-eng tunnels **(This command allows MPLS-TE globally)**

Configure on all core routers R1, R2, R3 and R4 with this command.

R1(config)# router ospf 1
R1(config)#network 0.0.0.0 0.0.0.0 area 0
R1(config-router)#**mpls traffic-eng area 0**
R1(config-router)#**mpls traffic-eng router-id loopback 0**

OSPF needs to recognize and use the OSPF Opaque LSA (type 10) to include the Traffic Engineering paths that will be configured. This command must also be configured on ALL core routers.

R1(config)# interface s0/0.12
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth 60
R1(config)# interface f0/0
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth 60
R1(config-if)# interface s0/0.13
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth (*we leave this option with the default*)

ALL interfaces that may carry tunnel traffic at any time must be configured with the "mpls traffic-eng tunnels" command on the interface as well. If this command is not issued at this level, the router will not participate in MPLS Traffic Engineering path calculation (PCALC).

ALL interfaces that may carry tunnel traffic at any time must be configured with the "ip rsvp bandwidth" command on the interface that could be a potential path. The router will not participate in MPLS Traffic Engineering path calculation (PCALC) from the head-end of the tunnel if not configured. The default value, if not specified, will be 75% of the "bandwidth" command configured on the interface. 1.544 Mbps is the default bandwith of any of the serial interfaces. RSVP will use this value to subtract any requested bandwidth by a tunnel.

The three configurations: "mpls traffic-eng tunnels"(global), "mpls traffic-eng tunnels"(interface), and "ip rsvp bandwith" must all be used on all core interfaces to participate in MPLS Traffic Engineering, even if there will be no tunnels built from the router itself.

On R2

R2(config)# mpls traffic-eng tunnels

R2(config)# router ospf 1
R2(config)#network 0.0.0.0 0.0.0.0 area 0
R2(config-router)#mpls traffic-eng area 0
R2(config-router)#mpls traffic-eng router-id loopback 0

R2(config)# interface s0/0.21
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth

R2(config-if)# interface s0/0.24
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth

On R3

R3(config)# mpls traffic-eng tunnels

R3(config)# router ospf 1
R3(config)#network 0.0.0.0 0.0.0.0 area 0
R3(config-router)#mpls traffic-eng area 0
R3(config-router)#mpls traffic-eng router-id loopback 0

R3(config)# interface s0/0.31
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth

R3(config-if)# interface s0/0.34
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth

On R4

R4(config)# mpls traffic-eng tunnels

R4(config)# router ospf 1
R4(config)#network 0.0.0.0 0.0.0.0 area 0
R4(config-router)#mpls traffic-eng area 0

```
R4(config-router)#mpls traffic-eng router-id loopback 0

R4(config)# interface s0/0.42
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth

R4(config)# interface f0/0
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth

R4(config-if)# interface s0/0.43
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth
```

Verifying basic traffic Engineering:

On R1

```
R1#sh mpls traffic-eng link-management igp-neighbors
Link ID::  Fa0/0
   Neighbor ID:  10.1.14.4 (area: ospf area 0, IP: 0.0.0.0)
Link ID::  Se0/0.12
   Neighbor ID:  2.2.2.2 (area: ospf area 0, IP: 10.1.12.2)
Link ID::  Se0/0.13
   Neighbor ID:  3.3.3.3 (area: ospf area 0, IP: 10.1.13.3)
```

MPLS traffic engineering has been correctly configured under the core interfaces or we would not be able to see neighbors.

```
R1#sh ip ospf database | be Type-10
            Type-10 Opaque Link Area Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum    Opaque ID
1.0.0.0         1.1.1.1         183         0x80000002 0x004347    0
1.0.0.0         2.2.2.2         148         0x80000002 0x00CCB9    0
1.0.0.0         3.3.3.3         112         0x80000002 0x004932    0
1.0.0.0         4.4.4.4         97          0x80000002 0x00CCE4    0
1.0.0.1         1.1.1.1         31          0x80000001 0x0038B6    1
1.0.0.1         2.2.2.2         139         0x80000001 0x0087EC    1
1.0.0.1         3.3.3.3         108         0x80000002 0x001C3E    1
1.0.0.1         4.4.4.4         93          0x80000002 0x00A50F    1
1.0.0.2         1.1.1.1         23          0x80000001 0x00AE77    2
```

The type 10 opaque LSA is used to identify all of the LSA's learned from the MPLS traffic-engineering topology. The highlighted paths all share the same link ID's which represents the tunnel and all of the LSA's learned from that tunnel.

R1#sh ip rsvp int

interface	allocated	i/f max	flow max	sub max
F0/0	0	**60K**	60K	0
Se0/0	0	0	0	0
Se0/0.12	0	60K	60K	0
Se0/0.13	0	**1158K**	**1158K**	0

RSVP has been configured under each of the core interfaces as well. The default-reserved bandwidth is 75% of the physical bandwidth. Default would be Se0/0.13 = 1544000 – 386000 (25%) = 1158K. Since we set the bandwidth to restrict to 60K on the other interfaces, it overrides the default.

Configure R1 s0/0.13 with "ip rsvp bandwidth 60" command before proceeding!!

Task 2

Configure MPLS-TE to establish basic connectivity from R1 to R4. R1 should not follow the default routing path. The path through R2 should be favored as a static path with the following attributes. Priority = 4, all traffic uses the tunnel to 4.4.4.4 and bandwidth = 30K.

Verify Current Routing Path:

On R1

```
R1#trace 4.4.4.4

Type escape sequence to abort.
Tracing the route to 4.4.4.4

  1 10.1.14.4 4 msec * 4 msec

R1#sh ip cef exact-route 1.1.1.1 4.4.4.4
1.1.1.1         -> 4.4.4.4        : FastEthernet0/0 (next hop 10.1.14.4)
```

The current path is currently R1-R4 through the F0/0 interface.

Change the Current Routing Path:

```
R1(config)# interface tunnel 0
R1(config-if)# ip unnumbered loopback 0
```

A good practice to not waste an ip address on the tunnel interface.

R1(config-if)# tunnel destination 4.4.4.4

The destination of the tunnel = the traffic-eng router-id of the last hop router

R1(config-if)# tunnel mode mpls traffic-eng

The tunnel mode must be changed from the default point-to-point tunnel mode, to mpls traffic engineering

R1(config-if)# tunnel mpls traffic-eng autoroute announce

This command ensures that all traffic is routed into the tunnel and announced into ospf updates.

R1(config-if)# tunnel mpls traffic-eng priority 4 4

The tunnel priority will use the SETUP priority(the first number) and compare with another tunnel's HOLD priority(the second number in the command). Both numbers are usually set to be the same to avoid confusion but may be different to establish true priority.

R1(config-if)# tunnel mpls traffic-eng bandwidth 30
R1(config-if)# tunnel mpls traffic-eng path-option 1 explicit name **SLOW_LINK**
R1(config-if)# exit

The tunnel should still remain down until the explicit path is created.

R1(config)# ip explicit-path name **SLOW_LINK** enable
R1(config)# next-address 10.1.12.2
R1(config)# next-address 10.1.24.4

The path should come up after the last hop is configured

Verify the Traffic Engineered path Information

On R1

```
R1#sh mpls traffic-eng tunnels brief
Signalling Summary:
    LSP Tunnels Process:           running
    RSVP Process:                  running
    Forwarding:                    enabled
    Periodic reoptimization:       every 3600 seconds, next in 2149 seconds
    Periodic auto-bw collection:   disabled
```

```
TUNNEL NAME                      DESTINATION      UP IF     DOWN IF    STATE/PROT
R1_t0                            4.4.4.4          -         Se0/0.12   up/up
```
Displayed 1 (of 1) heads, 0 (of 0) midpoints, 0 (of 0) tails

Notice that this is the "head" of the tunnel. It does not represent any other tunnel as a midpoint or a tail. The LSP Tunnel and RSVP processes show a running state.

Forwarding should be enabled or the top two processes do not matter.

Periodic reoptimazation allows for the tunnels to recalculate the best paths automatically over a specific amount of time (3600 seconds). There will be another update in 2149 seconds.

Periodic auto-bw collection is currently disabled and is used when automatically attempting to reroute based on utilization.

The tunnel name, "R1_t0", can sometimes be referenced in the command line. The destination of the tunnel is the TE Router-ID of the last hop TE router R4.

The "UP IF" is a term used to describe the Downstream peer(next hop router) from a Data Plane perspective. The "DOWN IF" is "Serial 0/0.12". The router output locks the field in this output to a certain size. In some cases, the interface name is too long for the space. The interface can be seen with the "show mpls traffic-eng tunnels" command.

The State and Protocol are treated as layer 1 and Layer 2 for the physical interface. Possible states are "*up/up*"- the physical line and the line protocol are operational , *up/down*- the physical layer is good for the tunnel but the line protocol has been broken as in the example of a sub-interface or physical interface failure. The tunnel interface is fine from a logical and physical layer view.

```
R1#sh mpls traffic-eng tunnels

Name: R1_t0                              (Tunnel0) Destination: 4.4.4.4
  Status:
    Admin: up        Oper: up     Path: valid       Signalling: connected

    path option 1, type explicit SLOW_LINK (Basis for Setup, path weight 128)

  Config Parameters:
    Bandwidth: 30       kbps (Global)  Priority: 4  4   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute:  enabled   LockDown: disabled  Loadshare: 30       bw-based
```

```
  auto-bw: disabled

 InLabel  :  -
 OutLabel : Serial0/0.12, 16
 RSVP Signalling Info:
      Src 1.1.1.1, Dst 4.4.4.4, Tun_Id 0, Tun_Instance 4
   RSVP Path Info:
     My Address: 1.1.1.1
     Explicit Route: 10.1.12.2 10.1.24.4 4.4.4.4
     Record Route:  NONE
     Tspec: ave rate=30 kbits, burst=1000 bytes, peak rate=30 kbits
   RSVP Resv Info:
     Record Route:  NONE
     Fspec: ave rate=30 kbits, burst=1000 bytes, peak rate=30 kbits
 History:
   Tunnel:
     Time since created: 2 hours, 17 minutes
     Time since path change: 2 hours, 15 minutes
   Current LSP:
     Uptime: 2 hours, 15 minutes
```

The name, destination have already been covered.

The "Admin: up" , "Oper: up,
Path: Valid, and Signalling: connected, make up extremly useful ouput for troubleshooting purposes. By the end of the labs, you should be able to use the "Path valid" and "Signalling" output to its' fullest intent.

The " path option 1" describes the number of the path-option declared by configuration that the tunnel has chosen. "type Explicit SLOW_LINK" was the explicit map configured earlier. The "(basis for Setup, setup weight 128)" is igp/te cost = 64 + 64 = 128. The Router uses the TE metric by default for path selection process and the te metric matches the igp metric by default.

The section Config Parameters section covers "Bandwidth: 30". This reflects the bandwidth required by the tunnel to be established in kbps and it is taken form the (Global) pool.

The "setup priority" is listed here for quick reference.

Tunnel Affinity is covered later but is noted here as a 32 bit value if needed.

"Metric Type is "TE" by default as covered in the explanation of the basis for setup.

"Autoroute- enable" was the chosen method to route traffic into the tunnel. The

"Lockdown- disabled" feature does not allow the tunnel to move from the current path and is disabled by default.

The "Load-share- disabled" can be used instead of specifying the bandwidth manually. It will load share the traffic between tunnels if more than one tunnel exist and has this feature enabled as well. There is further explanation of (bw-based) protocols later in the lesson. "auto-bw- disabled" can be used in conjunction with manual bandwidth statements but is disabled by default.

The "Inlabel" displays the label as it applies to the data plane. A label that arrives from an peer is swapped with the " OutLabel" that may be imposed initially, as in this case. The " Outlabel" displays the label that is forwarded toward the route origination point and specifies the interface (Serial 0/0.12) and the label (16) that is used to accomplish this task.

The "RSVP signalling Info:" displays the SouRCe and DeSTination of the tunnel RSVP configured tunnel. The tunnel ID is "0" and the "Tun_Instance" will change anytime the tunnel is torn down.

The "RSVP Path Info:" list the address of the router and the "Explicit Route" of the hop-by-hop path from R1 to R4 through R2. " Record Route" will be covered later. The Tspec , or Traffic specification is used to enforce policing on the traffic using the tunnel to the Tspec value specified in kbps. The Fspec or Flow specification is the value that will be used in RESerVation (RESV) messages by a receiver node to request QOS parameters.

The History of the tunnel can prove very useful in troubleshooting to establish patterns of success or failure. The "Tunnel:" is the time since the tunnel was created and how long since the last path change.

The "Current LSP:" output lists how long the current tunnel has been operational.

The "Prior LSP:" shows the path option in question and the "Removal Trigger" list what the potential problem or cause of the last LSP change or failure. The prior instance" is listed between the brackets after the path option.

```
R1#sh ip rsvp int
interface    allocated  i/f max  flow max    sub max
F0/0          0          60K      60K          0
Se0/0.12     30K         60K      60K          0
Se0/0         0          0         0           0
Se0/0.13      0          60K      60K          0
```

The interface through R2 is reserving the bandwidth as 30K.

Control and Data Plane:

On R1

R1#sh mpls ldp parameters
Protocol version: 1
No label generic region for downstream label distribution
Session hold time: 180 sec; keep alive interval: 60 sec
Discovery hello: holdtime: 15 sec; interval: 5 sec
Discovery targeted hello: holdtime: 90 sec; interval: 10 sec
Downstream on Demand max hop count: 255
Downstream on Demand Path Vector Limit: 255
LDP for targeted sessions
LDP initial/maximum backoff: 15/120 sec
LDP loop detection: off

R1 is performing "Downstream on Demand" learning of labels. This means that R1 will need to receive a label concerning a Downstream Network first before allocating a label for a given destination.

On R4

R4#sh mpls traffic-eng tunnels int

LSP Tunnel R1_t0 is signalled, connection is up
InLabel : Serial0/0.42, implicit-null
OutLabel : -

---text omitted---

R1 sends an RSVP "PATH message" that request for a label concerning the tunnel. The request passes through R2 and eventually reaches R4. Once the request is received for the tunnel, it is assigned a label of "implicit null" and will send this label to R2. The "OutLabel" represents the action that will be taken in the event it receives the "InLabel" context. In this case, this is the termination point so there should be no Outlabel Context.

On R2

R2#sh mpls traffic-eng tunnels int

LSP Tunnel R1_t0 is signalled, connection is up
InLabel : **Serial0/0.21, 16**
OutLabel : **Serial0/0.24, implicit-null**
RSVP Signalling Info:
Src 1.1.1.1, Dst 4.4.4.4, Tun_Id 0, Tun_Instance 28
RSVP Path Info:

```
    My Address: 10.1.12.2
    Explicit Route: 10.1.24.4 4.4.4.4
    Record Route:  NONE
    Tspec: ave rate=30 kbits, burst=1000 bytes, peak rate=30 kbits
  RSVP Resv Info:
    Record Route:  NONE
    Fspec: ave rate=30 kbits, burst=1000 bytes, peak rate=30 kbits

R2#sh mpls traffic-eng tun brie | b TUNNEL
TUNNEL NAME                DESTINATION    UP IF     DOWN IF   STATE/PROT
R1_t0                      4.4.4.4        Se0/0.21  Se0/0.24      up/up
Displayed 0 (of 0) heads, 1 (of 1) midpoints, 0 (of 0) tails


R2#sh mpls forwarding-table | b Local
Local  Outgoing    Prefix           Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id     switched   interface
16     Pop tag     1.1.1.1 0 [4]    0          Se0/0.24   point2point
```

Control Plane: R2 assignes a local tag of "16" for the tunnel when it receives the implicit null tag from R4. The tag could be different in your output.

Data Plane: R2 is waiting for label "16" on the InLabel interface s0/0.21 and forwarding the route with the implicit null tag (label "3") out of "OutLabel" int s0/0.24.

On R1

```
R1#sh mpls traffic-eng tunnels int s0/0.12

Name: R1_t0                          (Tunnel0) Destination: 4.4.4.4
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected

    path option 1, type explicit SLOW_LINK (Basis for Setup, path weight 128)

  Config Parameters:
    Bandwidth: 30       kbps (Global)  Priority: 4  4   Affinity: 0x0/0xFFFF

  Metric Type: TE (default)
    AutoRoute:  enabled   LockDown: disabled  Loadshare: 30       bw-based
    auto-bw: disabled

  InLabel  :  -
  OutLabel : Serial0/0.12, 16
```

```
  RSVP Signalling Info:
     Src 1.1.1.1, Dst 4.4.4.4, Tun_Id 0, Tun_Instance 28
   RSVP Path Info:
     My Address: 1.1.1.1
     Explicit Route: 10.1.12.2 10.1.24.4 4.4.4.4
     Record Route:  NONE
     Tspec: ave rate=30 kbits, burst=1000 bytes, peak rate=30 kbits
   RSVP Resv Info:
     Record Route:  NONE
     Fspec: ave rate=30 kbits, burst=1000 bytes, peak rate=30 kbits
  History:
   Tunnel:
     Time since created: 5 minutes, 43 seconds
     Time since path change: 4 minutes, 34 seconds
   Current LSP:
     Uptime: 4 minutes, 34 seconds
   Prior LSP:
     ID: path option 1 [6]
     Removal Trigger: label reservation removed
```

Control Plane: R1 learns the "16 label from R2.
Data Plane: Label "16" will be imposed if R1 receives a packet destined to use the tunnel.

```
R1#trace 4.4.4.4

Type escape sequence to abort.
Tracing the route to 4.4.4.4

  1 10.1.12.2 [MPLS: Label 16 Exp 0] 48 msec 28 msec 4 msec
  2 10.1.24.4 4 msec *  4 msec

R1#sh ip cef exact-route 1.1.1.1 4.4.4.4
1.1.1.1        -> 4.4.4.4       : Tunnel0 (next hop 0.0.0.0)
```

R1 uses the slower path due to the TE configuration on the R1 tunnel head.

Task 3

Configure a second tunnel (tunnel "1") on R1 that uses the same path option configured in Task 2. Provide an additional path option 2 MEDIUM_LINK on both tunnels. This path option should use the path through R3.

On R1

```
R1(config)#interface Tunnel0
R1(config-if)# tunnel mpls traffic-eng path-option 2 explicit name MEDIUM_LINK
```

We add the second path option on tunnel 0

```
R1(config)#interface Tunnel1
R1(config-if)# ip unnumbered Loopback0
R1(config-if)# tunnel destination 4.4.4.4
R1(config-if)# tunnel mode mpls traffic-eng
R1(config-if)# tunnel mpls traffic-eng autoroute announce
R1(config-if)# tunnel mpls traffic-eng priority 4 4
R1(config-if)# tunnel mpls traffic-eng bandwidth  30
R1(config-if)# tunnel mpls traffic-eng path-option 1 explicit name SLOW_LINK
R1(config-if)# tunnel mpls traffic-eng path-option 2 explicit name MEDIUM_LINK
```

The second tunnel is created with similar options for simplicity. The path will not come up until at least one of the explicit paths is configured.

```
R1(config)# ip explicit-path name MEDIUM_LINK
R1(cfg-ip-expl-path)# next-address 10.1.13.3
Explicit Path name MEDIUM_LINK:
    1: next-address 10.1.13.3
R1(cfg-ip-expl-path)# next-address 10.1.34.4
Explicit Path name MEDIUM_LINK:
    1: next-address 10.1.13.3
    2: next-address 10.1.34.4

R1(cfg-ip-expl-path)#end
```

The tunnel comes up immediately. 2 paths have been created. We need to see what tunnel is being used and analyze the constraints.

```
R1#sh mpls traffic-eng tunnels name R1_t0

Name: R1_t0                          (Tunnel0) Destination: 4.4.4.4
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected

    path option 1, type explicit SLOW_LINK (Basis for Setup, path weight 128)
    path option 2, type explicit MEDIUM_LINK

  Config Parameters:
    Bandwidth: 30       kbps (Global)  Priority: 4  4   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute:  enabled   LockDown: disabled  Loadshare: 30       bw-based
```

```
  auto-bw: disabled

 InLabel  :  -
 OutLabel : Serial0/0.12, 16
 RSVP Signalling Info:
     Src 1.1.1.1, Dst 4.4.4.4, Tun_Id 0, Tun_Instance 26
   RSVP Path Info:
    My Address: 1.1.1.1
    Explicit Route: 10.1.12.2 10.1.24.4 4.4.4.4

---text omitted---
```

The "Tunnel 0" interface uses the 1st path option since the tunnel priority does not come into play in this scenario and the bandwidth is available.

```
R1#sh mpls traffic-eng tunnels name-regexp ^R1_t1$
```

Yes….We can use regular expressions to match on names!!

```
Name: R1_t1                              (Tunnel1) Destination: 4.4.4.4
 Status:
   Admin: up         Oper: up     Path: valid       Signalling: connected

   path option 1, type explicit SLOW_LINK (Basis for Setup, path weight 128)
   path option 2, type explicit MEDIUM_LINK

 Config Parameters:
   Bandwidth: 30       kbps (Global)  Priority: 4  4   Affinity: 0x0/0xFFFF
   Metric Type: TE (default)
   AutoRoute:  enabled   LockDown: disabled  Loadshare: 30       bw-based
   auto-bw: disabled

 InLabel  :  -
 OutLabel : Serial0/0.12, 17
 RSVP Signalling Info:
     Src 1.1.1.1, Dst 4.4.4.4, Tun_Id 1, Tun_Instance 1
   RSVP Path Info:
    My Address: 1.1.1.1
    Explicit Route: 10.1.12.2 10.1.24.4 4.4.4.4

---text omitted---
```

"Tunnel 1" also uses the SLOW_LINK. There is enough bandwidth available and it is also the first option on this tunnel as well.

Task 4

Configure Tunnel "1 "to use 40 K of bandwidth and reoptomize the tunnels. Since RSVP bandwidth has been limited to 60K on every interface, view the changes that result in reoptimization of the tunnels.

On R1

```
R1(config)#interface tunnel 1
R1(config-if)#tunnel mpls traffic-eng bandwidth 40
```

We will view the effects of changing the bandwidth requirement of "Tunnel 1"

```
R1#sh mpls traffic-eng tunnels brie
Signalling Summary:
    LSP Tunnels Process:           running
    RSVP Process:                  running
    Forwarding:                    enabled
    Periodic reoptimization:       every 3600 seconds, next in 1585 seconds
    Periodic auto-bw collection:   disabled
TUNNEL NAME          DESTINATION      UP IF      DOWN IF    STATE/PROT
R1_t0                4.4.4.4          -          Se0/0.12   up/up
R1_t1                4.4.4.4          -          Se0/0.13   up/up
Displayed 2 (of 2) heads, 0 (of 0) midpoints, 0 (of 0) tails
```

The traffic that is currently using the SLOW_LINK may not change until reoptimization takes place (3600 seconds). In ths case, a changing of the resources has taken place which triggers te path change due to the constraint. Disabling timer frequency or events will have no effect.

```
R1#sh mpls traffic-eng tunnel role head source-id 1.1.1.1 name R1_t1

Name: R1_t1                          (Tunnel1) Destination: 4.4.4.4
  Status:
    Admin: up        Oper: up     Path: valid       Signalling: connected

    path option 2, type explicit MEDIUM_LINK (Basis for Setup, path weight 128)
    path option 1, type explicit SLOW_LINK

  Config Parameters:
    Bandwidth: 40       kbps (Global)  Priority: 4  4   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute:  enabled   LockDown: disabled  Loadshare: 40       bw-based
    auto-bw: disabled
```

```
InLabel  :  -
OutLabel : Serial0/0.13, 16
RSVP Signalling Info:
     Src 1.1.1.1, Dst 4.4.4.4, Tun_Id 1, Tun_Instance 3
  RSVP Path Info:
    My Address: 1.1.1.1
    Explicit Route: 10.1.13.3 10.1.34.4 4.4.4.4
    Record Route:  NONE
    Tspec: ave rate=40 kbits, burst=1000 bytes, peak rate=40 kbits
  RSVP Resv Info:
    Record Route:  NONE
    Fspec: ave rate=40 kbits, burst=1000 bytes, peak rate=40 kbits
History:
  Tunnel:
    Time since created: 5 minutes, 30 seconds
    Time since path change: 2 minutes, 6 seconds
  Current LSP:
    Uptime: 2 minutes, 6 seconds
    Selection: reoptimation
  Prior LSP:
    ID: path option 1 [1]
    Removal Trigger: configuration changed
    Last Error: PCALC:: Can't use link 0.0.0.0 on node 1.1.1.1
```

Notice how although the SLOW_LINK is the 1st option, the MEDIUM_PATH is chosen because the primary link only has a bandwitch of 60 kbps configured on the interface. The "Tunnel 0" interface has a requirement of 30 kbps. The Tunnel 1" interface has a requirement of 40 kbps and RSVP will reject the reservation on the primary path. The second path is unused and is chosen as the best for this tunnel. Also, the Current LSP information displays that the new LSP was due to reoptimization. The reoptimization is now automatic when a change is performed on the tunnel and does not necessarily need to be manually triggered.

<u>Task 5</u>

Configure the tunnels such that when shutdown manually, Tunnel"1" will use the primary path no matter which order the tunnels come up. Even during reoptimization.

<u>On R1</u>

```
R1(config)#int tun 0
R1(config-if)#shut
R1(config-if)#int tun 1
R1(config-if)#shut
```

R1(config)#int tunnel 1
R1(config-if)#tunnel mpls traffic-eng priority 3 4
% Setup priority (3) may not be higher than hold priority (4)
R1(config-if)#tunnel mpls traffic-eng priority 3 3

The priority is used to accomplish the task correctly. The hold priority cannot be higher than the setup priority. They are normally set to be identical.

R1(config)#int tun 0
R1(config-if)#no shut
R1(config-if)#
*Mar 1 00:48:26.939: %LINK-3-UPDOWN: Interface Tunnel0, changed state to up
*Mar 1 00:48:27.939: %LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel0, changed state to up

R1#sh mpls traffic-eng tunnel brie | b TUNNEL

TUNNEL NAME	DESTINATION	UP IF	DOWN IF	STATE/PROT
R1_t0	4.4.4.4	-	**Se0/0.12**	up/up
R1_t1	4.4.4.4	-	unknown	admin-down

Displayed 2 (of 2) heads, 0 (of 0) midpoints, 0 (of 0) tails

We bring up tunnel "0" first and make sure it is established. It uses the primary path through R2.

R1(config-if)#int tun 1
R1(config-if)#no shut

Verifying the Priority constraint:

On R1

R1#sh mpls traffic-eng tunnels name R1_t1 | include path option
path option 1, type explicit SLOW_LINK (Basis for Setup, path weight 128)
path option 2, type explicit MEDIUM_LINK
ID: path option 2 [16]

As we can see. Tunnel "1" gets established along the primary path. It bumps off Tunnel 0" in lieu of the better setup priority.

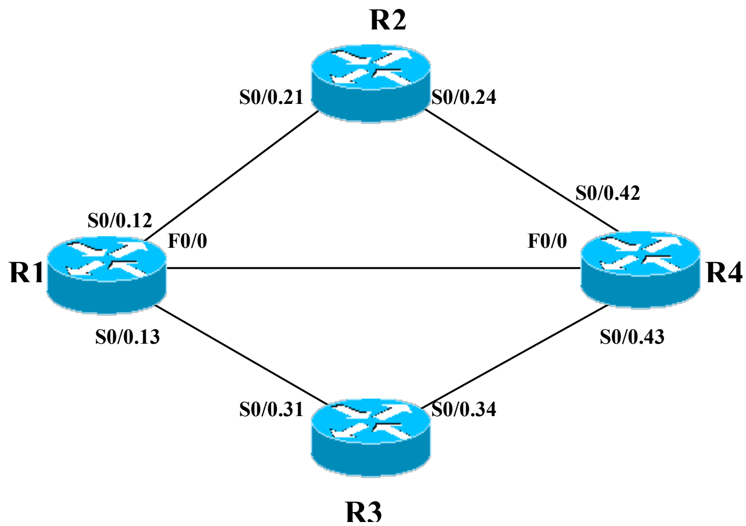
R1#sh mpls traffic-eng tunnels name R1_t0 | include path option
path option 2, type explicit MEDIUM_LINK (Basis for Setup, path weight 128)
path option 1, type explicit SLOW_LINK
ID: path option 1 [33]

Tunnel "0" is setup across the other path instead.

Task 6

Stop all routers in the console window and exit. Stop the Server and press the "Erase Start" launcher before proceeding.

Lab 2 – Dynamic Tunnels



Router	Interface	IP address
R1	Lo0 S0/0.12 S0/0.13 F0/0	1.1.1.1 /32 10.1.12.1 /24 10.1.13.1 /24 10.1.14.1 /24
R2	Lo0 S0/0.21 S0/0.23	2.2.2.2 /32 10.1.12.2 /24 10.1.24.2 /24
R3	Lo0 S0/0.31 S0/0.34	3.3.3.3/ 32 10.1.13.3 /24 10.1.34.3 /24
R4	Lo0 Lo1 F0/0 S0/0.42 S0/0.43	4.4.4.4 /32 44.4.4.4 /32 10.1.14.4 /24 10.1.24.4 /24 10.1.34.4 /24

Lab Setup:

The lab must have basic IGP and MPLS traffic Engineering configured to support tunnels. RSVP should use a bandwidth of 80 Kbps.

Task 1

Configure an OSPF with a process of "1" on all routers. The routers must be prepared to support MPLS-TE. Complete connectivity between the devices is required.

On R1

R1(config)# mpls traffic-eng tunnels

R1(config)# router ospf 1
R1(config)#network 0.0.0.0 0.0.0.0 area 0
R1(config-router)#mpls traffic-eng area 0
R1(config-router)#mpls traffic-eng router-id loopback 0

R1(config)# interface s0/0.12
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth 80

R1(config)# interface f0/0
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth 80

R1(config-if)# interface s0/0.13
R1(config-if)#bandwidth 768
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth 80

On R2

R2(config)# mpls traffic-eng tunnels

R2(config)# router ospf 1
R2(config)#network 0.0.0.0 0.0.0.0 area 0
R2(config-router)#mpls traffic-eng area 0
R2(config-router)#mpls traffic-eng router-id loopback 0

R2(config)# interface s0/0.21
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth 80

R2(config-if)# interface s0/0.24
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth 80

On R3

R3(config)# mpls traffic-eng tunnels

R3(config)# router ospf 1
R3(config)#network 0.0.0.0 0.0.0.0 area 0
R3(config-router)#mpls traffic-eng area 0
R3(config-router)#mpls traffic-eng router-id loopback 0

R3(config)# interface s0/0.31
R3(config-if)# **bandwidth 768**
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth 80

R3(config-if)# interface s0/0.34
R3(config-if)#**bandwidth 768**
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth 80

On R4

R4(config)# mpls traffic-eng tunnels

R4(config)# router ospf 1
R4(config)#network 0.0.0.0 0.0.0.0 area 0
R4(config-router)#mpls traffic-eng area 0
R4(config-router)#mpls traffic-eng router-id loopback 0

R4(config)# interface s0/0.42
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth 80

R4(config)# interface f0/0
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth 80

R4(config-if)# interface s0/0.43
R4(config-if)#**bandwidth 768**
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth 80

Verify the Configuration

On R1

R1#sh ip route ospf | i O

O 2.2.2.2 [110/65] via 10.1.12.2, 00:00:09, Serial0/0.12
O 4.4.4.4 [110/11] via 10.1.14.4, 00:00:09, FastEthernet0/0
O 10.1.24.0 [110/74] via 10.1.14.4, 00:00:09, FastEthernet0/0
O 10.1.34.0 [110/140] via 10.1.14.4, 00:00:09, FastEthernet0/0
O 44.4.4.4 [110/11] via 10.1.14.4, 00:00:09, FastEthernet0/0

R1#sh mpls int

Interface	IP	Tunnel	Operational
FastEthernet0/0	No	**Yes**	**Yes**
Serial0/0.12	No	**Yes**	**Yes**
Serial0/0.13	No	**Yes**	**Yes**

On R2

R2#sh ip route ospf | i O
O 1.1.1.1 [110/65] via 10.1.12.1, 00:02:07, Serial0/0.21
O 3.3.3.3 [110/195] via 10.1.24.4, 00:02:07, Serial0/0.24
O 4.4.4.4 [110/65] via 10.1.24.4, 00:02:07, Serial0/0.24
O 10.1.14.0 [110/74] via 10.1.24.4, 00:02:07, Serial0/0.24
O 10.1.13.0 [110/194] via 10.1.12.1, 00:02:07, Serial0/0.21
O 10.1.34.0 [110/194] via 10.1.24.4, 00:02:07, Serial0/0.24
O 44.4.4.4 [110/65] via 10.1.24.4, 00:02:07, Serial0/0.24

R2#sh mpls int

Interface	IP	Tunnel	Operational
Serial0/0.21	No	**Yes**	**Yes**
Serial0/0.24	No	**Yes**	**Yes**

On R3

R3#sh ip route ospf | i O
O 1.1.1.1 [110/131] via 10.1.13.1, 00:02:44, Serial0/0.31
O 2.2.2.2 [110/195] via 10.1.34.4, 00:02:44, Serial0/0.34
O 4.4.4.4 [110/131] via 10.1.34.4, 00:02:44, Serial0/0.34
O 10.1.14.0 [110/140] via 10.1.34.4, 00:02:44, Serial0/0.34
O 10.1.12.0 [110/194] via 10.1.13.1, 00:02:44, Serial0/0.31
O 10.1.24.0 [110/194] via 10.1.34.4, 00:02:44, Serial0/0.34
O 44.4.4.4 [110/131] via 10.1.34.4, 00:02:44, Serial0/0.34

R3#sh mpls int

Interface	IP	Tunnel	Operational
Serial0/0.31	No	**Yes**	**Yes**
Serial0/0.34	No	**Yes**	**Yes**

On R4

```
R4#sh ip route ospf | i O
O       1.1.1.1 [110/11] via 10.1.14.1, 00:03:11, FastEthernet0/0
O       2.2.2.2 [110/65] via 10.1.24.2, 00:03:11, Serial0/0.42
O       3.3.3.3 [110/131] via 10.1.34.3, 00:03:11, Serial0/0.43
O       10.1.13.0 [110/140] via 10.1.14.1, 00:03:11, FastEthernet0/0
O       10.1.12.0 [110/74] via 10.1.14.1, 00:03:11, FastEthernet0/0

R4#sh mpls int
Interface              IP            Tunnel      Operational
FastEthernet0/0        No            Yes         Yes
Serial0/0.42           No            Yes         Yes
Serial0/0.43           No            Yes         Yes
```

Task 2

Create a tunnel from R1 to R4 (Tunnel0) that automatically builds according to the current cost of the links. The signalled bandwidth should be 50K. Tunnel Priority should be setup of 1 and hold of 1. All traffic from any source that terminates to 4.4.4.4 should use the tunnel.

On R1

```
R1(config)#interface Tunnel0
R1(config-if)# ip unnumbered Loopback0
R1(config-if)# tunnel destination 4.4.4.4
R1(config-if)# tunnel mode mpls traffic-eng
R1(config-if)# tunnel mpls traffic-eng autoroute announce
R1(config-if)# tunnel mpls traffic-eng priority 1 1
R1(config-if)# tunnel mpls traffic-eng bandwidth  50
R1(config-if)# tunnel mpls traffic-eng path-option 1 dynamic
```

The path option uses the "dynamic" keyword instead of static to accomplish the task.

Verify the dynamic tunnel:

```
R1#show mpls traffic-eng tunnels up | s RSVP Path Info:
    RSVP Path Info:
      My Address: 10.1.14.1
      Explicit Route: 10.1.14.4 4.4.4.4
      Record Route:  NONE
      Tspec: ave rate=50 kbits, burst=1000 bytes, peak rate=50 kbits
```

The path direct to R4 is chosen as the best path with the necessary bandwidth available.

```
R1#sh mpls traffic-eng link-management admission-control f0/0
System Information::
    Tunnels Count:        1
    Tunnels Selected:     1
TUNNEL ID       UP IF    DOWN IF   PRIORITY    STATE          BW (kbps)
1.1.1.1 0_1     -        Fa0/0     1/1         Resv Admitted  50        RG

R1#sh ip rsvp int f0/0
interface     allocated   i/f max   flow max   sub max
Fa0/0         50K         80K       80K        0
```

30K more can be allocated on this interface.

Task 2

Configure a second tunnel called Tunnel "10 "to use 40 K of bandwidth. The priority should be identical to tunnel 0. Traffic destined for 44.4.4.4 should use this tunnel instead of tunnel "0". Policy-Based routing is not permitted.

On R1

```
R1(config)#interface Tunnel10
R1(config-if)# ip unnumbered Loopback0
R1(config-if)# tunnel destination 4.4.4.4
R1(config-if)# tunnel mode mpls traffic-eng
R1(config-if)# tunnel mpls traffic-eng priority 1 1
R1(config-if)# tunnel mpls traffic-eng bandwidth  40
R1(config-if)# tunnel mpls traffic-eng path-option 1 dynamic
R1(config-if)# exit
```

The bandwidth constraint has been raised to 40 to force a calculation to be made on which tunnel gets to use the best path. The interface has now been overutilized by 10K.

```
R1(config)# ip route 44.4.4.4 255.255.255.255 tunnel 10
```

This option can be used instead of routing traffic with autoroute announce

Verify Path Info:

```
R1#sh ip cef exact-route 1.1.1.1 44.4.4.4
1.1.1.1        -> 44.4.4.4       : Tunnel10 (attached)
```

R1#sh ip cef exact-route 1.1.1.1 4.4.4.4
1.1.1.1 -> 4.4.4.4 : **Tunnel0** (next hop 0.0.0.0)

The autoroute announce option on "Tunnel 0" allows ALL traffic that is destined for 4.4.4.4 to use tunnel 0. The static overrides this constraint by choosing to use "Tunnel 10" for routes to the 44.4.4.4 destination prefix.

R1#sh mpls traffic-eng tunnels name R1_t10 | section RSVP Path Info:
RSVP Path Info:
My Address: 1.1.1.1
Explicit Route: 10.1.12.2 10.1.24.4 4.4.4.4
Record Route: NONE
Tspec: ave rate=40 kbits, burst=1000 bytes, peak rate=40 kbits

The path through R2 was chosen due to the fact that Tunnel 0 was already established with a tunnel priority of "1" for setup and "1" for hold. "Tunnel 10" attempts to signal but the bandwidth constraint is not sufficient for the path to R4 directly. The tunnel reroutes all traffic through R2 since it is the next best path with a cost of 64.

Task 3

Tunnel 10 should take the path through R1, R3 and R4. Priority, Policy based Routing explicit paths or modifying OSPF cost is not permitted.

Viewing the current tunnel state:

R1#sh mpls traffic-eng link-management advertisements | sec Link ID:: 1
Link ID:: 1
Link IP Address: 10.1.12.1
IGP Neighbor: ID 2.2.2.2, IP 10.1.12.2
TE metric: 64
IGP metric: 64
Physical Bandwidth: 1544 kbits/sec
Res. Global BW: 80 kbits/sec
Res. Sub BW: 0 kbits/sec
Downstream::

---text omitted---

R1#sh mpls traffic-eng tunnels name R1_t10 | sec Config Parameters:
Config Parameters:
Bandwidth: 40 kbps (Global) Priority: 1 1 Affinity: 0x0/0xFFFF

Metric Type: TE (default)
AutoRoute: disabled LockDown: disabled Loadshare: 40 bw-based
auto-bw: disabled

TE metric is another cost that can be used instead of the IGP metric. By default, the tunnel TE metrics and also equals the IGP metric. (TE = IGP).

```
R1#show mpls traffic-eng link-management bandwidth-allocation | inc Admin. Weight:
   Admin. Weight:    10 (IGP)
   Admin. Weight:    64 (IGP)
   Admin. Weight:    130 (IGP)
```

```
R1#show mpls traffic-eng link-managemen interfaces s0/0.13 | incl Admin. Weight:
   Admin. Weight:    130 (IGP)
```

The cost of all 3 links from R1 to R4 are as follows: F0/0 = *10*, S0/0.12 = *64* and S0/0.13 = *80*.

On R1

```
R1(config)# interface s0/0.13
R1(config-subif)#mpls traffic-eng administrative-weight 1
```

```
R1(config)# interface tunnel 0
R1(config-if)#tunnel mpls traffic-eng path-selection metric igp
R1(config-if)# end
R1#mpls traffic-eng reoptimize
```

```
R1#sh mpls traffic-eng tunnels name R1_t0 | sec  RSVP Path Info:
  RSVP Path Info:
   My Address: 10.1.14.1
   Explicit Route: 10.1.14.4 4.4.4.4
   Record Route:  NONE
   Tspec: ave rate=50 kbits, burst=1000 bytes, peak rate=50 kbits
```

```
R1#sh mpls traffic-eng tunnels name R1_t10 | be RSVP Path Info:
  RSVP Path Info:
   My Address: 1.1.1.1
   Explicit Route: 10.1.13.3 10.1.34.4 4.4.4.4
```

```
R1#show mpls traffic-eng link-managemen interfaces s0/0.13 | incl Admin. Weight:
Admin. Weight:    1 (configured)
```

```
R1#sh mpls traffic-eng tunnels name R1_t10 | s Status
 Status:
  Admin: up       Oper: up    Path: valid      Signalling: connected
```

path option 1, type dynamic **(Basis for Setup, path weight 2)**

Tunnel "0" uses the path through the lowest IGP cost which still uses the path to R4 directly. Tunnel"10" uses the path that reflects the lowest TE metric of "2".

Task 4

Tunnel 0 should take the path through R2 to R4. Priority, Policy based Routing, explicit paths modifying OSPF cost or TE metric is not permitted.

Viewing the current tunnel paths:

R1#trace 4.4.4.4

Type escape sequence to abort.
Tracing the route to 4.4.4.4

1 10.1.14.4 8 msec * 4 msec

R1#trace 44.4.4.4

Type escape sequence to abort.
Tracing the route to 44.4.4.4

1 10.1.13.3 [MPLS: Label 16 Exp 0] 52 msec 4 msec 24 msec
2 10.1.34.4 24 msec * 8 msec

Currently, the path for 4.4.4.4 is across the f0/0 interface directly to R4. This tunnel routes ALL traffic behind it to Tunnel 0.

On R1

R1(config-if)#int s0/0.12
R1(config-if)#mpls traffic-eng attribute-flags **0xAD**

R1(config)# int Tunnel 0
R1(config-if)#tunnel mpls traffic-eng affinity 0x1 mask **0x3**

On R2

R2(config-if)#int s0/0.21

R2(config-if)#mpls traffic-eng attribute-flags **0xAD**
R2(config-if)#int s0/0.24
R2(config-if)#mpls traffic-eng attribute-flags **0xAD**

Choosing the hexadecimal notation must be consistent if used. The other method is to provide the binary equivalent of 0x00110001 (0x31). This also must be consistent along the path. The tunnel should tear down when modifying the parameters. A reoptimization should not be necessary.

1) Only a "1" and a "1" will yield a "1" in the output when the Attribute Flag or Affinity is ANDED with the mask.

2) Affinity + Mask = Attribute flag + Mask

Tunnel 0

Affinity	**0x00000001**	**(0x1)**
Mask	**0x00000011**	**(0x3)**
Result	**00000001**	

Path 1

Attribute flag	**0x10110110**	**(0xB6)**
Mask	**0x00000011**	**(0x3)**
Result	**00000010 (no match)**	

Path 2

Attribute flag	**0x10101101**	**(0xAD)**
Mask	**0x00000011**	**(0x3)**
Result	**00000001 (match)**	

Path 1 was chosen to show a comparison of values when computing Affinity as a constraint. PATH 1 should not be used since it does not match the Result from the Tunnel Affinity and Mask. Only Path 2 matches and will be considered in path selection.

Verifying the tunnel path:

On R1

```
R1#sh mpls traffic-eng tunnels name R1_t0

Name: R1_t0                          (Tunnel0) Destination: 4.4.4.4
  Status:
    Admin: up        Oper: up     Path: valid      Signalling: connected

    path option 1, type dynamic (Basis for Setup, path weight 128)

  Config Parameters:
    Bandwidth: 50       kbps (Global)  Priority: 1  1   Affinity: 0x1/0x3
    Metric Type: IGP (interface)
    AutoRoute:  enabled   LockDown: disabled  Loadshare: 50       bw-based
```

```
  auto-bw: disabled

 InLabel  :  -
 OutLabel : Serial0/0.12, 16
 RSVP Signalling Info:
      Src 1.1.1.1, Dst 4.4.4.4, Tun_Id 0, Tun_Instance 19
   RSVP Path Info:
     My Address: 1.1.1.1
     Explicit Route: 10.1.12.2 10.1.24.4 4.4.4.4
     Record Route:  NONE
     Tspec: ave rate=50 kbits, burst=1000 bytes, peak rate=50 kbits
   RSVP Resv Info:
     Record Route:  NONE
     Fspec: ave rate=50 kbits, burst=1000 bytes, peak rate=50 kbits
 History:
   Tunnel:
     Time since created: 5 minutes, 40 seconds
     Time since path change: 2 minutes, 23 seconds
   Current LSP:
     Uptime: 2 minutes, 23 seconds
   Prior LSP:
     ID: path option 1 [12]
     Removal Trigger: configuration changed
```

The tunnel is restored and shown to have switched to the path where the values match. The Prior LSP shows that the configuration caused the switch to occur.

Task 5

Allow Tunnel "0" to record the bandwidth every 30 seconds. Configure Tunnel"0" to have a tunnel bandwidth greater than "0" as a configuration under the tunnel interface without explicitly configuring it. The configuration should remain for at least 5 minutes. Tunnel 10 should be limited to no more than 90K of bandwidth at any time.

```
R1#sh mpls traffic-eng tunnels name R1_t0 | sec Config Parameters:

 Config Parameters:
   Bandwidth: 50      kbps (Global)  Priority: 1  1   Affinity: 0x1/0x3
   Metric Type: IGP (interface)
   AutoRoute:  enabled   LockDown: disabled  Loadshare: 50      bw-based
   auto-bw: disabled
```

The auto-bandwidth feature is currently disabled and is the focus of this task.

On R1

R1(config)#mpls traffic-eng auto-bw timers frequency 30
R1(config)# int tunnel 0
R1(config-if)#tunnel mpls traffic-eng auto-bw **frequency 300**

After 300 seconds,
R1(config)# int tunnel 10
R1(config-if)#tunnel mpls traffic-eng auto-bw **max-bw 90**

Auto bandwidth can be configured with the following parameters:

"mpls traffic-eng auto-bw frequency" – This command when configured globally will record the bandwidth on ALL TE interfaces in a prescribed amount of time in seconds.

"tunnel mpls traffic-eng auto-bw"- This command changes optional parameters for auto-bw calculation. The default is 86400 seconds (1 day).

collect-bw **– only states what the bandwidth would be when a change takes place.**

frequency**- changes the time when the bandwidth is used from the recorded data on the TE interface. The router will automatically configure the requested bandwidth under the apprpriate tunnel.**

max-bw **– maximizes how much bandwidth can be allocated for a tunnel when the frequency timer expires.**

min-bw **- expresses the minimum bandwidth required for allocation to the tunnel when the frequency timer expires.**

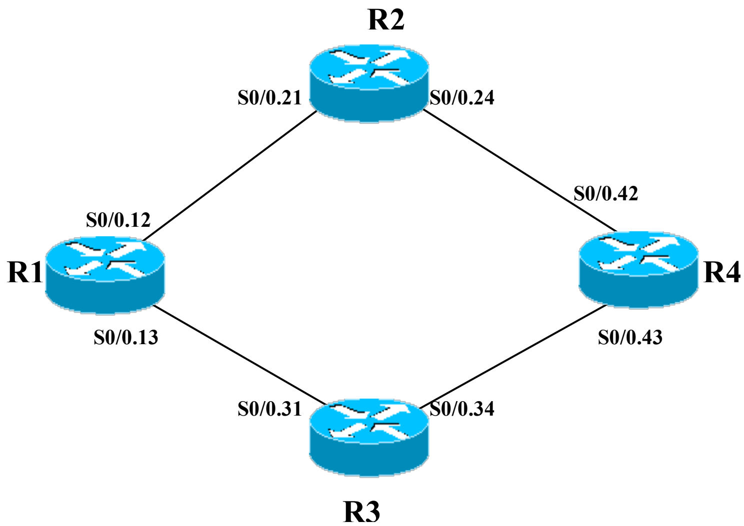
Verify the configuration:

R1#sh mpls traffic-eng tunnels name R1_t0 | sec Config Parameters:

```
 Config Parameters:
   Bandwidth: 50      kbps (Global)  Priority: 1  1   Affinity: 0x1/0x3
   Metric Type: IGP (interface)
   AutoRoute:  enabled   LockDown: disabled  Loadshare: 50      bw-based
   auto-bw: (300/283) 0  Bandwidth Requested: 50
```

The 300 is the mpls frequency interval. The 285 represents the time left before a Bandwidth Calculation is retrived from the buffer. The first "0" is the current bandwidth that has been measured. The " Bandwidth Requested: 50" will post a number when the frequency timer expires and resets.

Lab 3 – Load Balancing



Router	Interface	IP address
R1	Lo0 S0/0.12 S0/0.13	1.1.1.1 /32 10.1.12.1 /24 10.1.13.1 /24
R2	Lo0 S0/0.21 S0/0.23	2.2.2.2 /32 10.1.12.2 /24 10.1.24.2 /24
R3	Lo0 S0/0.31 S0/0.34	3.3.3.3/ 32 10.1.13.3 /24 10.1.34.3 /24
R4	Lo0 Lo44 S0/0.42 S0/0.43	4.4.4.4 /32 44.44.44.44/32 10.1.24.4 /24 10.1.34.4 /24

Lab Setup:

All Frame-Relay links are connected in a point-to-point manner.

Task 1

Configure an OSPF with a process of "1" on all routers. The routers must be prepared to support MPLS-TE. Complete connectivity between the devices is required. All RSVP interfaces should be configured to allow 30K worth of traffic on ANY path.

On R1

R1(config)# mpls traffic-eng tunnels

R1(config)# router ospf 1
R1(config)#network 0.0.0.0 0.0.0.0 area 0
R1(config-router)#mpls traffic-eng area 0
R1(config-router)#mpls traffic-eng router-id loopback 0

R1(config)# interface s0/0.12
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth 30

R1(config)# interface f0/0
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth 30

R1(config-if)# interface s0/0.13
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth 30

On R2

R2(config)# mpls traffic-eng tunnels

R2(config)# router ospf 1
R2(config)#network 0.0.0.0 0.0.0.0 area 0
R2(config-router)#mpls traffic-eng area 0
R2(config-router)#mpls traffic-eng router-id loopback 0

R2(config)# interface s0/0.21
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth 30

R2(config-if)# interface s0/0.24
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth 30

On R3

R3(config)# mpls traffic-eng tunnels

R3(config)# router ospf 1
R3(config)#network 0.0.0.0 0.0.0.0 area 0
R3(config-router)#mpls traffic-eng area 0
R3(config-router)#mpls traffic-eng router-id loopback 0

R3(config)# interface s0/0.31
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth 30

R3(config-if)# interface s0/0.34
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth 30

On R4

R4(config)# mpls traffic-eng tunnels

R4(config)# router ospf 1
R4(config)#network 0.0.0.0 0.0.0.0 area 0
R4(config-router)#mpls traffic-eng area 0
R4(config-router)#mpls traffic-eng router-id loopback 0

R4(config)# interface s0/0.42
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth 80

R4(config)# interface f0/0
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth 80

R4(config-if)# interface s0/0.43
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth 80

Verify the Configuration

On R1

R1#sh ip route ospf | i O
O 2.2.2.2 [110/65] via 10.1.12.2, 00:00:09, Serial0/0.12
O 4.4.4.4 [110/11] via 10.1.14.4, 00:00:09, FastEthernet0/0
O 10.1.24.0 [110/74] via 10.1.14.4, 00:00:09, FastEthernet0/0

O 10.1.34.0 [110/140] via 10.1.14.4, 00:00:09, FastEthernet0/0
O 44.4.4.4 [110/11] via 10.1.14.4, 00:00:09, FastEthernet0/0

R1#sh mpls int

Interface	IP	Tunnel	Operational
FastEthernet0/0	No	**Yes**	**Yes**
Serial0/0.12	No	**Yes**	**Yes**
Serial0/0.13	No	**Yes**	**Yes**

On R2

R2#sh ip route ospf | i O
O 1.1.1.1 [110/65] via 10.1.12.1, 00:02:07, Serial0/0.21
O 3.3.3.3 [110/195] via 10.1.24.4, 00:02:07, Serial0/0.24
O 4.4.4.4 [110/65] via 10.1.24.4, 00:02:07, Serial0/0.24
O 10.1.14.0 [110/74] via 10.1.24.4, 00:02:07, Serial0/0.24
O 10.1.13.0 [110/194] via 10.1.12.1, 00:02:07, Serial0/0.21
O 10.1.34.0 [110/194] via 10.1.24.4, 00:02:07, Serial0/0.24
O 44.4.4.4 [110/65] via 10.1.24.4, 00:02:07, Serial0/0.24

R2#sh mpls int

Interface	IP	Tunnel	Operational
Serial0/0.21	No	**Yes**	**Yes**
Serial0/0.24	No	**Yes**	**Yes**

On R3

R3#sh ip route ospf | i O
O 1.1.1.1 [110/131] via 10.1.13.1, 00:02:44, Serial0/0.31
O 2.2.2.2 [110/195] via 10.1.34.4, 00:02:44, Serial0/0.34
O 4.4.4.4 [110/131] via 10.1.34.4, 00:02:44, Serial0/0.34
O 10.1.14.0 [110/140] via 10.1.34.4, 00:02:44, Serial0/0.34
O 10.1.12.0 [110/194] via 10.1.13.1, 00:02:44, Serial0/0.31
O 10.1.24.0 [110/194] via 10.1.34.4, 00:02:44, Serial0/0.34
O 44.4.4.4 [110/131] via 10.1.34.4, 00:02:44, Serial0/0.34

R3#sh mpls int

Interface	IP	Tunnel	Operational
Serial0/0.31	No	**Yes**	**Yes**
Serial0/0.34	No	**Yes**	**Yes**

On R4

R4#sh ip route ospf | i O
O 1.1.1.1 [110/11] via 10.1.14.1, 00:03:11, FastEthernet0/0
O 2.2.2.2 [110/65] via 10.1.24.2, 00:03:11, Serial0/0.42

```
O       3.3.3.3 [110/131] via 10.1.34.3, 00:03:11, Serial0/0.43
O       10.1.13.0 [110/140] via 10.1.14.1, 00:03:11, FastEthernet0/0
O       10.1.12.0 [110/74] via 10.1.14.1, 00:03:11, FastEthernet0/0

R4#sh mpls int
Interface              IP          Tunnel      Operational
FastEthernet0/0        No          Yes         Yes
Serial0/0.42           No          Yes         Yes
Serial0/0.43           No          Yes         Yes
```

Task 2

Create two new dynamic tunnels from R1 to R4 (4.4.4.4). The tunnels should have the attributes listed in the table below and should use autoroute announce. Tunnel "0" should be created first and should be torn down as a result of the second tunnel being built, however Tunnel 0 should not participate in reoptimization if the path through R3 becomes available.

Tunnel	Bandwidth	Priority (Setup/Hold)
Tunnel 0	30	7/7
Tunnel 1	30	6/6

Configuring the tunnels:

On R1

```
R1(config)#interface Tunnel0
R1(config-if)# ip unnumbered Loopback0
R1(config-if)# tunnel destination 4.4.4.4
R1(config-if)# tunnel mode mpls traffic-eng
R1(config-if)# tunnel mpls traffic-eng autoroute announce
R1(config-if)# tunnel mpls traffic-eng priority 7 7
R1(config-if)# tunnel mpls traffic-eng bandwidth  30
R1(config-if)# tunnel mpls traffic-eng path-option 1 dynamic lockdown

R1(config-if)#interface Tunnel1
R1(config-if)# ip unnumbered Loopback0
R1(config-if)# tunnel destination 4.4.4.4
R1(config-if)# tunnel mode mpls traffic-eng
R1(config-if)# tunnel mpls traffic-eng autoroute announce
R1(config-if)# tunnel mpls traffic-eng priority 6 6
```

```
R1(config-if)# tunnel mpls traffic-eng bandwidth  30
R1(config-if)# tunnel mpls traffic-eng path-option 1 dynamic
```

The lockdown option is used when a tunnel establishes connectivity and does not participate when an "mpls traffic-eng reoptimize" command is issued.

Verifying the Tunnel Setup:

```
R1#sh mpls traffic-eng tunnels

Name: R1_t0                              (Tunnel0) Destination: 4.4.4.4
  Status:
    Admin: up         Oper: up      Path: valid       Signalling: connected

    path option 1,(LOCKDOWN) type dynamic (Basis for Setup, path weight 128)

---text omitted---
    RSVP Path Info:
      My Address: 1.1.1.1
      Explicit Route: 10.1.12.2 10.1.24.4 4.4.4.4

Name: R1_t1                              (Tunnel1) Destination: 4.4.4.4
  Status:
    Admin: up         Oper: up      Path: valid       Signalling: connected

---text omitted---

    RSVP Path Info:
      My Address: 1.1.1.1
      Explicit Route: 10.1.13.3 10.1.34.4 4.4.4.4
```

Tunnel "1" has a better priority and gets first path choice which is R1-R3-R4. Tunnel "0 has taken the path through R2.

```
R1(config)#int tunnel 1
R1(config-if)#shut

R1#mpls traffic-eng reoptimize

Name: R1_t0                              (Tunnel0) Destination: 4.4.4.4
  Status:
    Admin: up         Oper: up      Path: valid       Signalling: connected

path option 1,(LOCKDOWN) type dynamic (Basis for Setup, path weight 128)

      Explicit Route: 10.1.12.2 10.1.24.4 4.4.4.4
```

After shutting down Tunnel"1", the bandwidth becomes available on the best path. When forcing the reoptimize feature, the bandwidth does not get reallocated for Tunnel"0" due to the "(LOCKDOWN)" status.

```
R2(config)#int s0/0.24
R2(config-subif)#shut
```

Here, we are simulating a failure by shutting down the interface somewhere in the path of Tunnel "0".

```
Name: R1_t0                    (Tunnel0) Destination: 4.4.4.4
  Status:
    Admin: up       Oper: up     Path: valid      Signalling: connected

    path option 1,(LOCKDOWN) type dynamic (Basis for Setup, path weight 128)

  Config Parameters:
    Bandwidth: 30      kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute:  enabled   LockDown: enabled   Loadshare: 30      bw-based
    auto-bw: disabled

  InLabel  :  -
  OutLabel : Serial0/0.13, 16
  RSVP Signalling Info:
       Src 1.1.1.1, Dst 4.4.4.4, Tun_Id 0, Tun_Instance 6
    RSVP Path Info:
      My Address: 1.1.1.1
      Explicit Route: 10.1.13.3 10.1.34.4 4.4.4.4

---text omitted---

History:
    Tunnel:
      Time since created: 7 minutes, 8 seconds
      Time since path change: 29 seconds
    Current LSP:
      Uptime: 29 seconds
      Selection: reoptimation
    Prior LSP:
      ID: path option 1 [5]
      Removal Trigger: path error
```

The path of the tunnel has changed to be what it would not reoptimize earlier. If a true failure takes place along the path, the tunnel will attempt to find a better path.

Task 3

A tunnel must be created from R1 to R4. Ensure that 2 times as many packets go down the path through R3 compared to 1 packet through R2 . The route should be recorded in MPLS-TE.

Viewing the current tunnel paths:

Be sure to "no shut" tunnel "1" and the interface on R2 before continuing to the next lab.

On R1

```
R1#sh mpls traffic-eng autoroute 4.4.4.4
MPLS TE autorouting enabled
  destination 4.4.4.4 has 2 tunnels
    Tunnel0 (load balancing metric 66666666, nexthop 4.4.4.4)
    Tunnel1 (load balancing metric 66666666, nexthop 4.4.4.4)
```

The load balancing is calculated based on the tunnel bandwidth in kbp. (1,000,000 / 30 = 33,333.333). When this total is multiplied by itself, it equals the load sharing based on the output which = 66666666. Both paths have equal cost between them.

```
R1#sh ip route ospf | sec 44.4.4.4
O       4.4.4.4 [110/129] via 0.0.0.0, 00:02:06, Tunnel0
                [110/129] via 0.0.0.0, 00:02:06, Tunnel1
```

The cost to the 44.44.44.44 network prefix is currently 129. Cost = (64) between R1 and R2 + (64) Cost between R2 and R4 + Loopback = 1 (64+64+1 = 129)

```
R1(config)#int tun 0
R1(config-if)#tunnel mpls traffic-eng autoroute metric relative -5
```

```
R1#sh mpls traffic-eng autoroute
MPLS TE autorouting enabled
  destination 4.4.4.4 has 2 tunnels
    Tunnel0 (load balancing metric 66666666, nexthop 4.4.4.4, relative metric -5)
    Tunnel1 (load balancing metric 66666666, nexthop 4.4.4.4)
```

```
R1#sh ip route 44.4.4.4
Routing entry for 44.4.4.4/32
  Known via "ospf 1", distance 110, metric 124, type intra area
  Routing Descriptor Blocks:
  * directly connected, via Tunnel0
      Route metric is 124, traffic share count is 1
```

When the packet hits the end of the tunnel, 5 metric units are subtracted before counting each hop after. (64+64 – 5 = 124). In addition to the "*relative*" option, the "*absoute*" option will make the cost for everything after the tunnel to be the same no matter how many hops it takes to reach the destination through the tunnel. This option only works when ISIS is used as the IGP.

```
R1(config)#int tun 0
R1(config-if)#tunnel mpls traffic-eng load-share 30
R1(config-if)#int tun 1
R1(config-if)#tunnel mpls traffic-eng load-share 60

R1#sh mpls traffic-eng autoroute
MPLS TE autorouting enabled
  destination 4.4.4.4 has 2 tunnels
    Tunnel1 (load balancing metric 33333333, nexthop 4.4.4.4)
    Tunnel0 (load balancing metric 66666666, nexthop 4.4.4.4, relative metric -5
```

The load share can use any set of numbers to display the load balancing requested. 2 to 1 can also be represented by "200" and "100" respectively. The numbers may change but the load balancing is the same. The only difference is when the autoroute begins to load balance in relation to the bandwidth used.

Viewing the Tunnel prior to Route Record:

On R1

```
R1#sh mpls traffic-eng tunnels name R1_t0 | sec RSVP Path Info:
    RSVP Path Info:
      My Address: 1.1.1.1
      Explicit Route: 10.1.12.2 10.1.24.4 4.4.4.4
      Record Route:  NONE
      Tspec: ave rate=30 kbits, burst=1000 bytes, peak rate=30 kbits
```

On Tunnel 1:

```
R1(config)# interface tunnel 1
R1(config-if)#tunnel mpls traffic-eng record-route
R1(config-if)#end
R1# mpls traffic-enf reoptimize
```

Viewing the Tunnel after Route Record enabled:

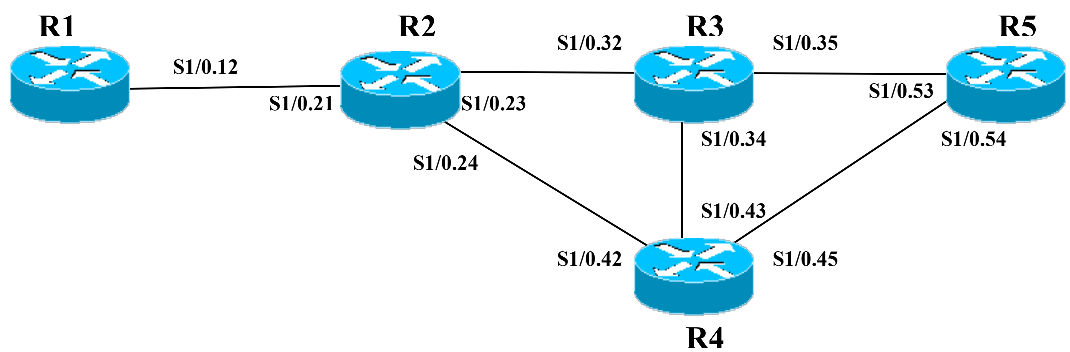
On R4

```
R4#sh mpls traffic-eng tunnels name R1_t0 | sec RSVP Path Info:
  RSVP Path Info:
    My Address: 4.4.4.4
    Explicit Route:  NONE
    Record Route:  NONE
    Tspec: ave rate=30 kbits, burst=1000 bytes, peak rate=30 kbits

R4#sh mpls traffic-eng tunnels name R1_t1 | sec RSVP Path Info:
  RSVP Path Info:
    My Address: 4.4.4.4
    Explicit Route:  NONE
    Record Route: 10.1.34.3 10.1.13.1
    Tspec: ave rate=30 kbits, burst=1000 bytes, peak rate=30 kbits
```

The Route Record option was only used on Tunnel "1". The route has been recorded and displayed on R4 as being valid.

Lab 4 –Fast reroute backup tunnels



Router	Interface	IP address
R1	Lo0 S1/0.12	1.1.1.1 /32 10.1.12.1 /24
R2	Lo0 S1/0.21 S1/0.23 S1/0.24	2.2.2.2 /32 10.1.12.2 /24 10.1.23.2/ 24 10.1.24.2 /24
R3	Lo0 S1/0.32 S1/0.34 S1/0.35	3.3.3.3/ 32 10.1.23.3 /24 10.1.34.3. 24 10.1.35.3 /24
R4	Lo0 S1/0.42 S1/0.43 S1/0.45	4.4.4.4 /32 10.1.24.4 /24 10.1.34.4 /24 10.1.45.4 /24
R5	Lo0 S1/0.53 S1/0.54	5.5.5.5 /32 10.1.35.5 /24 10.1.45.5/ 24

Lab Setup:

All Frame-Relay links are connected in a point-to-point manner. The labels ranges should be manipulated for clarity.

Task 1

Configure an OSPF with a process of "1" on all routers. The routers must be prepared to support MPLS-TE. Complete connectivity between the devices is required. All RSVP interfaces should be configured to allow default bandwidth utilization.

On R1

R1(config)# mpls traffic-eng tunnels
R1(config)# mpls label range 100 199

R1(config)# router ospf 1
R1(config)#network 0.0.0.0 0.0.0.0 area 0
R1(config-router)#mpls traffic-eng area 0
R1(config-router)#mpls traffic-eng router-id loopback 0

R1(config)# interface S1/0.12
R1(config-if)# mpls traffic-eng tunnels
R1(config-if)# ip rsvp bandwidth

On R2

R2(config)# mpls traffic-eng tunnels
R2(config)# mpls label range 200 299

R2(config)# router ospf 1
R2(config)#network 0.0.0.0 0.0.0.0 area 0
R2(config-router)#mpls traffic-eng area 0
R2(config-router)#mpls traffic-eng router-id loopback 0

R2(config)# interface S1/0.21
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth

R2(config)# interface S1/0.23
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth

R2(config-if)# interface S1/0.24
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth

On R3

R3(config)# mpls traffic-eng tunnels

```
R3(config)# mpls label range 300 399

R3(config)# router ospf 1
R3(config)#network 0.0.0.0 0.0.0.0 area 0
R3(config-router)#mpls traffic-eng area 0
R3(config-router)#mpls traffic-eng router-id loopback 0

R3(config)# interface S1/0.32
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth

R3(config-if)# interface S1/0.34
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth

R3(config)# interface S1/0.35
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth
```

On R4

```
R4(config)# mpls traffic-eng tunnels
R4(config)# mpls label range 400 499

R4(config)# router ospf 1
R4(config)#network 0.0.0.0 0.0.0.0 area 0
R4(config-router)#mpls traffic-eng area 0
R4(config-router)#mpls traffic-eng router-id loopback 0

R4(config)# interface S1/0.42
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth

R4(config-if)# interface S1/0.43
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth

R4(config)# interface S1/0.45
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth
```

On R5

```
R5(config)# mpls traffic-eng tunnels
R5(config)# mpls label range 500 599
```

```
R5(config)# router ospf 1
R5(config)#network 0.0.0.0 0.0.0.0 area 0
R5(config-router)#mpls traffic-eng area 0
R5(config-router)#mpls traffic-eng router-id loopback 0

R5(config)# interface S1/0.52
R5(config-if)# mpls traffic-eng tunnels
R5(config-if)# ip rsvp bandwidth

R5(config-if)# interface S1/0.53
R5(config-if)# mpls traffic-eng tunnels
R5(config-if)# ip rsvp bandwidth
```

Verify the Configuration

On R1

```
R1#sh ip route ospf | i O
O    2.2.2.2 [110/65] via 10.1.12.2, 00:00:53, Serial1/0.12
O    3.3.3.3 [110/129] via 10.1.12.2, 00:00:53, Serial1/0.12
O    4.4.4.4 [110/129] via 10.1.12.2, 00:00:53, Serial1/0.12
O    5.5.5.5 [110/193] via 10.1.12.2, 00:00:53, Serial1/0.12
O    10.1.24.0 [110/128] via 10.1.12.2, 00:00:53, Serial1/0.12
O    10.1.23.0 [110/128] via 10.1.12.2, 00:00:53, Serial1/0.12
O    10.1.45.0 [110/192] via 10.1.12.2, 00:00:53, Serial1/0.12
O    10.1.35.0 [110/192] via 10.1.12.2, 00:00:53, Serial1/0.12
O    10.1.34.0 [110/192] via 10.1.12.2, 00:00:53, Serial1/0.12

R1#sh mpls int
Interface        IP        Tunnel    Operational
Serial1/0.12     No        Yes       Yes
```

On R2

```
R2#sh ip route ospf | i O
O    1.1.1.1 [110/65] via 10.1.12.1, 00:01:33, Serial1/0.21
O    3.3.3.3 [110/65] via 10.1.23.3, 00:01:33, Serial1/0.23
O    4.4.4.4 [110/65] via 10.1.24.4, 00:01:33, Serial1/0.24
O    5.5.5.5 [110/129] via 10.1.24.4, 00:01:33, Serial1/0.24
O    10.1.45.0 [110/128] via 10.1.24.4, 00:01:33, Serial1/0.24
O    10.1.35.0 [110/128] via 10.1.23.3, 00:01:33, Serial1/0.23
O    10.1.34.0 [110/128] via 10.1.24.4, 00:01:33, Serial1/0.24

R2#sh mpls int
Interface        IP        Tunnel    Operational
Serial1/0.21     No        Yes       Yes
```

```
Serial1/0.24       No        Yes       Yes
Serial1/0.23       No        Yes       Yes
```

On R3

```
R3#sh ip route ospf | i O
O      1.1.1.1 [110/129] via 10.1.23.2, 00:01:56, Serial1/0.32
O      2.2.2.2 [110/65] via 10.1.23.2, 00:01:56, Serial1/0.32
O      4.4.4.4 [110/65] via 10.1.34.4, 00:01:56, Serial1/0.34
O      5.5.5.5 [110/65] via 10.1.35.5, 00:01:56, Serial1/0.35
O      10.1.12.0 [110/128] via 10.1.23.2, 00:01:56, Serial1/0.32
O      10.1.24.0 [110/128] via 10.1.34.4, 00:01:56, Serial1/0.34
O      10.1.45.0 [110/128] via 10.1.35.5, 00:01:56, Serial1/0.35
```

```
R3#sh mpls int
Interface          IP        Tunnel    Operational
Serial1/0.34       No        Yes       Yes
Serial1/0.35       No        Yes       Yes
Serial1/0.32       No        Yes       Yes
```

On R4

```
R4#sh ip route ospf | i O
O      1.1.1.1 [110/129] via 10.1.24.2, 00:02:25, Serial1/0.42
O      2.2.2.2 [110/65] via 10.1.24.2, 00:02:25, Serial1/0.42
O      3.3.3.3 [110/65] via 10.1.34.3, 00:02:25, Serial1/0.43
O      5.5.5.5 [110/65] via 10.1.45.5, 00:02:25, Serial1/0.45
O      10.1.12.0 [110/128] via 10.1.24.2, 00:02:25, Serial1/0.42
O      10.1.23.0 [110/128] via 10.1.34.3, 00:02:25, Serial1/0.43
O      10.1.35.0 [110/128] via 10.1.45.5, 00:02:25, Serial1/0.45
```

```
R4#sh mpls int
Interface          IP        Tunnel    Operational
Serial1/0.42       No        Yes       Yes
Serial1/0.43       No        Yes       Yes
Serial1/0.45       No        Yes       Yes
```

On R5

```
R5#sh ip route ospf | i O
O      1.1.1.1 [110/193] via 10.1.45.4, 00:02:59, Serial1/0.54
O      2.2.2.2 [110/129] via 10.1.45.4, 00:02:59, Serial1/0.54
O      3.3.3.3 [110/65] via 10.1.35.3, 00:02:59, Serial1/0.53
O      4.4.4.4 [110/65] via 10.1.45.4, 00:02:59, Serial1/0.54
O      10.1.12.0 [110/192] via 10.1.45.4, 00:02:59, Serial1/0.54
O      10.1.24.0 [110/128] via 10.1.45.4, 00:02:59, Serial1/0.54
```

```
O       10.1.23.0 [110/128] via 10.1.35.3, 00:02:59, Serial1/0.53
O       10.1.34.0 [110/128] via 10.1.45.4, 00:02:59, Serial1/0.54

R5#sh mpls int
Interface         IP        Tunnel    Operational
Serial1/0.53      No        Yes       Yes
Serial1/0.54      No        Yes       Yes
```

Task 2

Create a new static tunnel from R1 to R5 (5.5.5.5). Tunnel "0" should have the attributes listed in the table below and should use autoroute announce. Create a second tunnel on R2 with the same characteristics as Tunnel "0" that protects the link between R2 and R3. The static path for the newly formed tunnel"1" on R2 should be through the S1/0.23 next hop.

Tunnel	Bandwidth	Priority (Setup/Hold)
Tunnel 0	30	7/7

Configuring the tunnels:

On R1

```
R1(config)#interface Tunnel0
R1(config-if)# ip unnumbered Loopback0
R1(config-if)# tunnel destination 5.5.5.5
R1(config-if)# tunnel mode mpls traffic-eng
R1(config-if)# tunnel mpls traffic-eng autoroute announce
R1(config-if)# tunnel mpls traffic-eng priority 7 7
R1(config-if)# tunnel mpls traffic-eng bandwidth  30
R1(config-if)# tunnel mpls traffic-eng path-option 1 explicit name TST
R1(config-if)# tunnels mpls traffic-eng fast-reroute
R1(config-if)#exit
```

Fast Reroute enables protection of the tunnel from the Head End side. The default features of Fast Reroute is "link" and "node" protection and will be described as the lab moves forward.

```
R1(config)# ip explicit path name TST enable
R1(cfg-ip-expl-path) next-address 10.1.12.2
R1(cfg-ip-expl-path) next-address 10.1.23.3
R1(cfg-ip-expl-path) next-address 10.1.35.5
```

The tunnel should come up and use the path through R2.

R1#trace 5.5.5.5

Type escape sequence to abort.
Tracing the route to 5.5.5.5

 1 10.1.12.2 [MPLS: Label 200 Exp 0] 72 msec 40 msec 36 msec
 2 10.1.23.3 [MPLS: Label 300 Exp 0] 20 msec 20 msec 24 msec
 3 10.1.35.5 20 msec * 28 msec

The packet is labeled across the correct interface.

On R2

R2(config)#interface Tunnel1
R2(config-if)# ip unnumbered Loopback0
R2(config-if)# **tunnel destination 3.3.3.3**
R2(config-if)# tunnel mode mpls traffic-eng
R2(config-if)# tunnel mpls traffic-eng autoroute announce
R2(config-if)# tunnel mpls traffic-eng bandwidth 30
R2(config-if)# tunnel mpls traffic-eng priority 7 7
R2(config-if)# tunnel mpls traffic-eng path-option 1 **explicit name TST-BAK**
R2(config-if)#exit

R2(config)# ip **explicit-path name TST-BAK** enable
R2(cfg-ip-expl-path)# next-address 10.1.24.4
R2(cfg-ip-expl-path)# next-address 10.1.34.3

The tunnel destination should always be the next hop router on the other side of the protected link when performing "Link Protection". The exact static path has been typed in for clarity, however, the use of the "exclude" command to prevent certain next hops from being used could have been a valid option.

R2(cfg-ip-expl-path)#interface s1/0.23
R2(config-subif)# mpls traffic-eng backup-path Tunnel 1

The interface that is being protected must be configured to allow the backup tunnel to be used. This form of a tunnel is known as "Link Protection".

Verify the Link Protected Fast Reroute Tunnel:

On R2

R2#sh mpls traffic-eng fast-reroute database
Headend frr information:
Protected tunnel In-label Out intf/label FRR intf/label Status

LSP midpoint frr information:
LSP identifier In-label Out intf/label FRR intf/label Status
1.1.1.1 0 [4] 200 Se1/0.23:300 **Tu1**:300 **ready**

The fastreroute tunnel has already signalled the fast-reroute information to this node. It now understands what actions to take incase of a failure by leveraging RSVP signalling as a reroute mechanism.

We simulate a failure on the link between R2 and R4!!

R2(config)#int s1/0.23
R2(config-subif)# shut

R2#sh mpls traffic-eng fast-reroute database
Headend frr information:
Protected tunnel In-label Out intf/label FRR intf/label Status

LSP midpoint frr information:
LSP identifier In-label Out intf/label FRR intf/label Status
1.1.1.1 0 [4] 200 Se1/0.23:300 Tu1:300 **active**

The route goes active and the traffic has a slight disruption of service.

On R1

R1#trace 5.5.5.5

Type escape sequence to abort.
Tracing the route to 5.5.5.5

1 10.1.12.2 [MPLS: Label 200 Exp 0] 64 msec 32 msec 28 msec
2 10.1.24.4 [MPLS: Labels **400/300** Exp 0] 24 msec 24 msec 20 msec
3 10.1.34.3 [MPLS: Label 300 Exp 0] 28 msec 16 msec 36 msec
4 10.1.35.5 28 msec * 36 msec

The trace is still operational from R1. However, the path has clearly changed and is taking the backup route. Label "400" has also been added as the outer tag in addition to label "300" which represents the original tag that will be used by R3.

R1#sh mpls traffic-eng tunnels | sec History:

History:
Tunnel:
Time since created: 8 minutes, 12 seconds
Time since path change: 8 minutes, 11 seconds
Number of LSP IDs (Tun_Instances) used: 1
Current LSP:
Uptime: 1 minutes, 12 seconds
Last Error: PCALC:: Explicit path has unknown address, 10.1.23.3

We also verify that the last LSP was indeed torn down due to a Path CALCulation error when the interface was shutdown.

Restore the broken Link on R2:

On R2

```
R2(config)#int s1/0.23
R2(config-subif)#no shut
```

Task 3

Create a tunnel named Tunnel "2" on R2. The attributes should be similar to that of tunnel on R1. The tunnel should use a fast reroute tunnel that bypasses the failure no matter what point in the path the failure takes place.

Configuring the tunnel:

On R2

```
R2(config)#interface Tunnel2
R2(config-if)#ip unnumbered Loopback0
R2(config-if)#tunnel destination 5.5.5.5
R2(config-if)#tunnel mode mpls traffic-eng
R2(config-if)#tunnel mpls traffic-eng autoroute announce
R2(config-if)#tunnel mpls traffic-eng path-option 1 explicit name TST-BAK-2

R2(config-if)#ip explicit-path name TST-BAK-2
R2(cfg-ip-expl-path)# next-address 10.1.24.4
R2(cfg-ip-expl-path)# next-address 10.1.45.5

R2(cfg-ip-expl-path)#int s1/0.23
R2(config-subif)#mpls traffic-eng backup-path Tunnel 2
```

Here the tunnel destination represents the loopback interface of R5 instead of R3. In this case, "Node-Protection" requires that the termination be at least one hop beyond the next hop router. (referred to as NNHOP). The interface is again protected but by a different tunnel.

Verifying Tunnel operation:

R2#sh mpls traffic-eng fast-reroute database
Headend frr information:
Protected tunnel In-label Out intf/label FRR intf/label Status

LSP midpoint frr information:
LSP identifier In-label Out intf/label FRR intf/label Status
1.1.1.1 0 [26] 201 Se1/0.23:301 **Tu2**:implicit-nul **ready**

The tunnel is ready but the tunnel it is taking has changed to be tunnel "2". The router will always see a path that protects a NODE as being superior. This type of protection would protect anywhere in the path, not just merely a LINK.

R2(config)#int s1/0.23
R2(config-subif)#shut

Simulating the error again.

R2#sh mpls traffic-eng fast-reroute database
Headend frr information:
Protected tunnel In-label Out intf/label FRR intf/label Status

LSP midpoint frr information:
LSP identifier In-label Out intf/label FRR intf/label Status
1.1.1.1 0 [26] 201 Se1/0.23:301 Tu2:implicit-nul active

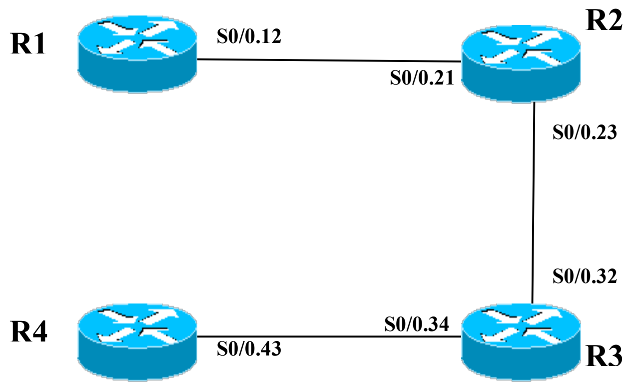
The route is now active to reroute traffic onto the different path specified.

R1#trace 5.5.5.5

Type escape sequence to abort.
Tracing the route to 5.5.5.5

1 10.1.12.2 [MPLS: Label 201 Exp 0] 412 msec 32 msec 20 msec
2 10.1.24.4 [MPLS: Label 401 Exp 0] 32 msec 20 msec 16 msec
3 10.1.45.5 28 msec * 24 msec

Lab 5 –MPLS QOS



Router	Interface	IP address
R1	Lo0 S0/0.12	1.1.1.1 /32 10.1.12.1 /24
R2	Lo0 S0/0.21 S0/0.23	2.2.2.2 /32 10.1.12.2 /24 10.1.23.2/ 24
R3	Lo0 S0/0.32 S0/0.34	3.3.3.3/ 32 10.1.23.3 /24 10.1.34.3. 24
R4	Lo0 S0/0.43	4.4.4.4 /32 10.1.34.4 /24

Lab Setup:

The lab has all frame relay PVC's configured in a point-to-point manner.

Task 1

Configure an OSPF with a process of "1" on all routers. The R2, R3 and R4 routers must be prepared to support MPLS-TE. All RSVP interfaces should be configured to allow a global bandwidth utilization of 100K.

On R1

```
R1(config)# router ospf 1
R1(config)#network 0.0.0.0 0.0.0.0 area 0
```

On R2

```
R2(config)# mpls traffic-eng tunnels
R2(config)# mpls label range 200 299

R2(config)# router ospf 1
R2(config)#network 0.0.0.0 0.0.0.0 area 0
R2(config-router)#mpls traffic-eng area 0
R2(config-router)#mpls traffic-eng router-id loopback 0

R2(config)# interface S0/0.23
R2(config-if)# mpls traffic-eng tunnels
R2(config-if)# ip rsvp bandwidth 100
```

On R3

```
R3(config)# mpls traffic-eng tunnels
R3(config)# mpls label range 300 399

R3(config)# router ospf 1
R3(config)#network 0.0.0.0 0.0.0.0 area 0
R3(config-router)#mpls traffic-eng area 0
R3(config-router)#mpls traffic-eng router-id loopback 0

R3(config)# interface S0/0.32
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth 100

R3(config-if)# interface S0/0.34
R3(config-if)# mpls traffic-eng tunnels
R3(config-if)# ip rsvp bandwidth 100
```

On R4

```
R4(config)# mpls traffic-eng tunnels
R4(config)# mpls label range 400 499

R4(config)# router ospf 1
R4(config)#network 0.0.0.0 0.0.0.0 area 0
R4(config-router)#mpls traffic-eng area 0
R4(config-router)#mpls traffic-eng router-id loopback 0
```

R4(config-if)# interface S0/0.43
R4(config-if)# mpls traffic-eng tunnels
R4(config-if)# ip rsvp bandwidth 100

Verify the Configuration

On R1

R1#sh ip route ospf | i O
O 2.2.2.2 [110/65] via 10.1.12.2, 00:00:39, Serial0/0.12
O 3.3.3.3 [110/129] via 10.1.12.2, 00:00:39, Serial0/0.12
O 4.4.4.4 [110/193] via 10.1.12.2, 00:00:39, Serial0/0.12
O 10.1.23.0 [110/128] via 10.1.12.2, 00:00:39, Serial0/0.12
O 10.1.34.0 [110/192] via 10.1.12.2, 00:00:39, Serial0/0.12

On R2

R2#sh ip route ospf | i O
O 1.1.1.1 [110/65] via 10.1.12.1, 00:00:50, Serial0/0.21
O 3.3.3.3 [110/65] via 10.1.23.3, 00:00:50, Serial0/0.23
O 4.4.4.4 [110/129] via 10.1.23.3, 00:00:50, Serial0/0.23
O 10.1.34.0 [110/128] via 10.1.23.3, 00:00:50, Serial0/0.23

R2#sh mpls int

Interface	IP	Tunnel	Operational
Serial0/0.23	No	**Yes**	**Yes**

On R3

R3#sh ip route ospf | i O
O 1.1.1.1 [110/129] via 10.1.23.2, 00:01:11, Serial0/0.32
O 2.2.2.2 [110/65] via 10.1.23.2, 00:01:11, Serial0/0.32
O 4.4.4.4 [110/65] via 10.1.34.4, 00:01:11, Serial0/0.34
O 10.1.12.0 [110/128] via 10.1.23.2, 00:01:11, Serial0/0.32

R3#sh mpls int

Interface	IP	Tunnel	Operational
Serial0/0.32	No	**Yes**	**Yes**
Serial0/0.34	No	**Yes**	**Yes**

On R4

R4#sh ip route ospf | i O
O 1.1.1.1 [110/193] via 10.1.34.3, 00:02:14, Serial0/0.43

```
O       2.2.2.2 [110/129] via 10.1.34.3, 00:02:14, Serial0/0.43
O       3.3.3.3 [110/65] via 10.1.34.3, 00:02:14, Serial0/0.43
O       10.1.12.0 [110/192] via 10.1.34.3, 00:02:14, Serial0/0.43
O       10.1.23.0 [110/128] via 10.1.34.3, 00:02:14, Serial0/0.43

R4#sh mpls int
Interface          IP        Tunnel    Operational
Serial0/0.43       No        Yes       Yes
```

Task 2

Create a tunnel called "Tunnel"0" from R2 to R4. This tunnel must have the following characteristics: setup and hold of "1", bandwidth of 40K, dynamic setup.

On R2

```
R2(config)#int tunnel 0
R2(config-if)#ip unnumbered lo0
R2(config-if)#tunnel destination 4.4.4.4
R2(config-if)#tunnel mode mpls traffic-eng
R2(config-if)#tunnel mpls traffic-eng priority 1 1
R2(config-if)#tunnel mpls traffic-eng bandwidth 40
R2(config-if)#tunnel mpls traffic-eng path-option 1 dynamic
```

Verify the tunnel:

```
R2#sh ip rsvp reservation
To          From        Pro  DPort  Sport  Next Hop     I/F       Fi  Serv   BPS
4.4.4.4     2.2.2.2     0    0      1      10.1.23.3    Se0/0.23  SE LOAD 40K

R2#sh mpls traffic-eng link-management admission-control
System Information::
   Tunnels Count:       1
   GMPLS Tunnels Count:  0
   Tunnels Selected:     1
TUNNEL ID         UP IF    DOWN IF    PRIORITY   STATE           BW (kbps)
2.2.2.2 0_1       -        Se0/0.23   1/1        Resv Admitted   40          RG

R2#sh mpls traffic-eng link-management advertisements | b Res
Res. Global BW:       100 kbits/sec
   Res. Sub BW:         0 kbits/sec
   Downstream::
                                        Global Pool      Sub Pool
                                        --------------   ----------
```

Reservable Bandwidth[0]:	**100**	0 kbits/sec
Reservable Bandwidth[1]:	**60**	0 kbits/sec
Reservable Bandwidth[2]:	60	0 kbits/sec
Reservable Bandwidth[3]:	60	0 kbits/sec
Reservable Bandwidth[4]:	60	0 kbits/sec
Reservable Bandwidth[5]:	60	0 kbits/sec
Reservable Bandwidth[6]:	60	0 kbits/sec
Reservable Bandwidth[7]:	60	0 kbits/sec

Attribute Flags: 0x00000000

The global pool has been allowed up to 100 kbps. Due to the tunnel priority, all other priorites have also adjusted the available bandwidth to be 60.

Task 3

Create a tunnel called "Tunnel"1" that has similar characteristics to that of tunnel "0". The only required change in the configuration will be to allow for bandwidth guarantees to be possible for certain tunnel priorities above "3". The guarantee for Tunnel "1" should be 25K."

Configuring DS-TE

On R2

R2(config)#int tunnel 1
R2(config-if)# ip unnumbered Loopback0
R2(config-if)# tunnel destination 4.4.4.4
R2(config-if)# tunnel mode mpls traffic-eng
R2(config-if)# tunnel mpls traffic-eng autoroute announce
R2(config-if)# tunnel mpls traffic-eng priority 4 4
R2(config-if)# tunnel mpls traffic-eng bandwidth **sub-pool 25**
R2(config-if)# tunnel mpls traffic-eng path-option 1 dynamic

The Tunnel Head-End can choose between "bandwidth" or "subpools" as an option but not both. Sub-pools allow for more granular usage of RSVP bandwidth reservations. Traffic that is routed down a tunnel can still experience congestion. The goal is to maintain some kind of banwidth management that fits into the current QOS model.

R2(config)# interface s0/0.23
R2(config-subif)#ip rsvp bandwidth 100 **sub-pool 40**

ON R3

R3(config)# interface s1/0.32
R3(config-subif)#ip rsvp bandwidth 100 **sub-pool 40**

R3(config-subif)# interface s1/0.34
R3(config-subif)#ip rsvp bandwidth 100 **sub-pool 40**

ON R4

R4(config)# interface s1/0.43
R4(config-subif)#ip rsvp bandwidth 100 **sub-pool 40**

The entire path must have subpools available in RSVP for ANY tunnel that uses sub-pools as an option to operate correctly. R4 does not need the configuration as it represents the last hop. The configuration is there for completeness.

Verify the change:

On R2

```
R2#sh mpls traffic-eng link-management advertisements | b Res
Res. Global BW:      100 kbits/sec
   Res. Sub BW:        40 kbits/sec
   Downstream::

                                          Global Pool      Sub Pool
                                          --------------   ----------
     Reservable Bandwidth[0]:                    100            40 kbits/sec
     Reservable Bandwidth[1]:                     60            40 kbits/sec
     Reservable Bandwidth[2]:                     60            40 kbits/sec
     Reservable Bandwidth[3]:                     60            40 kbits/sec
     Reservable Bandwidth[4]:                     35            15 kbits/sec
     Reservable Bandwidth[5]:                     35            15 kbits/sec
     Reservable Bandwidth[6]:                     35            15 kbits/sec
     Reservable Bandwidth[7]:                     35            15 kbits/sec
   Attribute Flags:      0x00000000
```

The tunnel reserves the bandwidth that has been reserved for the sub-pools as well as the global pool. The same does not hold true for the global reservations. They can only reserve bandwidth from the Global Pool only. The output above also shows how the bandwidth from a certain priority will reserve from all priorities that are inferior to it.

Task 4

All pings will be generated from R1 with a TOS of 184, 96 or 32. Create a service-policy on R2 that matches on a value other than DSCP or IPP. No filters are permitted. A previous marking must be matched. Create the same policy on R3 to verify the markings.

Configure QOS to match and mark packets:

On R2

R2(config)#class-map TST-5
R2(config-cmap)#match mpls experimental 5
R2(config-cmap)#exit

R2(config)#class-map TST-3
R2(config-cmap)#match mpls experimental 3
R2(config-cmap)#exit

We can match the experimental field due to the fact that the 3 most significant bits in the TOS field are automatically copied into the experimental field.

R2(config)#policy-map MARK-TST
R2(config-pmap)#class TST-5
R2(config-pmap-c)#priority percent 20
R2(config-pmap-c)#exit
R2(config-pmap)#class TST-3
R2(config-pmap-c)#bandwidth percent 10
R2(config-pmap-c)#exit

R2(config)#int s1/0
R2(config-subif)#service-policy output MARK-TST

On R3

R3(config)#class-map TST-5
R3(config-cmap)#match mpls experimental 5
R3(config-cmap)#exit

R3(config)#class-map TST-3
R3(config-cmap)#match mpls experimental 3
R3(config-cmap)#exit

R3(config)#policy-map MARK-TST
R3(config-pmap)#class TST-5

```
R3(config-pmap-c)#set mpls experimental 5
R3(config-pmap-c)#exit
R3(config-pmap)#class TST-3
R3(config-pmap-c)#set mpls experimental 3
R3(config-pmap-c)#exit

R3(config)#int s1/0.32
R3(config-subif)#service-policy input MARK-TST
```

This policy is only created to verify the marking is being performed by R2

Verify the markings:

On R1

```
R1#ping
Protocol [ip]:
Target IP address: 4.4.4.4
Repeat count [5]: 150
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 1.1.1.1
Type of service [0]: 160
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 150, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!
Success rate is 100 percent (150/150), round-trip min/avg/max = 4/18/72 ms

R1#ping
Protocol [ip]:
Target IP address: 4.4.4.4
Repeat count [5]: 99
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 1.1.1.1
Type of service [0]: 96
```

```
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 99, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (99/99), round-trip min/avg/max = 8/21/356 ms
```

On R2

```
R2#sh policy-map int s0/0

 Serial1/0

  Service-policy output: MARK-TST

    queue stats for all priority classes:
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 150/16200

    Class-map: TST-5 (match-all)
      150 packets, 15600 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: mpls experimental topmost 5
      Priority: 20% (308 kbps), burst bytes 7700, b/w exceed drops: 0

   Class-map: TST-3 (match-all)
      99 packets, 10296 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: mpls experimental topmost 3
      Priority: 10% (154 kbps), burst bytes 3850, b/w exceed drops: 0
---text omitted---
```

The packets were marked and placed in the proper queues.

On R3

```
R3#sh policy-map int
 Serial1/0
```

Service-policy input: MARK-TST

Class-map: TST-5 (match-all)
  150 packets, 16200 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
  Match: mpls experimental topmost 5
  QoS Set
    mpls experimental imposition 5
      Packets marked 150

Class-map: TST-3 (match-all)
  99 packets, 10692 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
  Match: mpls experimental topmost 3
  QoS Set
    mpls experimental imposition 3
      Packets marked 99

---text omitted---

The packets are verified on R3 and have followed the policy to have them remarked with the same values.

Task 5

Modify the traffic at Experimental value of 3 to be remarked to 1 on Tunnel "1", MPLS TE bandwidth should only be flooded when traffic is measured rising at 20% and 60 %. The bandwidth info should also be flooded when traffic decreases at 40% and 80 %.

On R2

R2(config)#policy-map MARK-TST
R2(config-pmap)#class TST-3
R2(config-pmap-c)#set mpls experimental topmost 1

The mpls experimental bit cannot be set on an outbound policy with the "imposition" keyword, it must be used with the "topmost" keyword instead.

On R3

R3(config)#class-map REMARK-TST
R3(config-cmap)#match mpls exp 1

R3(config-cmap)#policy-map MARK-TST
R3(config-pmap)#class REMARK-TST
R3(config-pmap-c)#set mpls exper 1

Verify the remarking:

On R1

```
R1#ping
Protocol [ip]:
Target IP address: 4.4.4.4
Repeat count [5]: 101
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 1.1.1.1
Type of service [0]: 96
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 101, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (101/101), round-trip min/avg/max = 4/19/60 ms

R2#sh policy-map int

---text omitted---

Class-map: TST-3 (match-all)
    200 packets, 20800 bytes
    5 minute offered rate 0 bps, drop rate 0 bps
    Match: mpls experimental topmost 3
    Priority: 10% (154 kbps), burst bytes 3850, b/w exceed drops: 0

    QoS Set
      mpls experimental topmost 1
        Packets marked 101
```

The packets were successfully matched and remarked to a lower QOS value.

On R3

R3#sh policy-map int

Class-map: REMARK-TST (match-all)
  101 packets, 10908 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
  Match: mpls experimental topmost 1
  QoS Set
   mpls experimental imposition 1
    Packets marked 101

R3 verifies the change indeed was remarked

Adjusting Thresholds:

On R2

R2# sh mpls traffic-eng link-management bandwidth-allocation | s Downstream
Downstream Global Pool Bandwidth Information (kbits/sec):

KEEP PRIORITY	BW HELD	BW TOTAL HELD	BW LOCKED	BW TOTAL LOCKED
0	0	0	0	0
1	0	0	40	40
2	0	0	0	40
3	0	0	0	40
4	0	0	0	40
5	0	0	0	40
6	0	0	0	40
7	0	0	0	40

Downstream Sub Pool Bandwidth Information (kbits/sec):

KEEP PRIORITY	BW HELD	BW TOTAL HELD	BW LOCKED	BW TOTAL LOCKED
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	**25**	**25**
5	0	0	0	**25**
6	0	0	0	**25**
7	0	0	0	**25**

The bandwidth is allocated to the subpool. Notice how the "MAX Res Sub BW" field has been changed to 40k which is the reserved bandwidth for the tunnel and the "BW Descriptors" field still has 1 descriptor but also shows that the descriptor includes the 1 subpool configured. This feature allows the tunnel to take advantage of a Modular QOS Policy that has LLQ enabled for priority bandwidth. The bandwidth that is provisioned in the Policy will utilize 100% of the subpool bandwidth out of the interface that also has the sub pool

command towards the destination.

R2# sh mpls traffic-eng link-management bandwidth-allocation | s Up | Down Thre
Up Thresholds: 15 30 45 60 75 80 85 90 95 96 97 98 99 100 (default)
Down Thresholds: 100 99 98 97 96 95 90 85 80 75 60 45 30 15 (default)

The thresholds show when the bandwidth information will be flooded into TE calculations for either UP % or Down %.

On R2

R2(config-if)#mpls traffic-eng flooding thresholds down 40 80
R2(config-if)#mpls traffic-eng flooding thresholds up 20 60

The default thresholds have many changes imposed at different levels. If we wish for a value to remain, the adjustment shoud contain the original value.

R2# sh mpls traffic-eng link-management bandwidth-allocation | s Up | Down Thre
Up Thresholds: **20 60**
Down Thresholds: **80 40**

The other thresholds are gone but what we have configured has remained.