
# FACULTAD DE TECNOLOGÍA INFORMÁTICA

# INGENIERÍA EN SISTEMAS INFORMÁTICOS


## Redes y Teleprocesamiento

## Practico 02 - MPLS

**Docentes:** Niell, Carlos.

**Alumno:** Di Domenico, Nicolás.

**Comisión:** 5° A.

**Turno:** Noche.

**Año:** 2025.

**Fecha:** 15/11/2025.

## Índice

# 1 Introducción

En este trabajo práctico se implementa una red MPLS (Multiprotocol Label Switching) sobre una maqueta de tres routers Cisco. El objetivo es observar cómo MPLS se apoya en el enrutamiento IP tradicional y en CEF para conmutar paquetes utilizando etiquetas en lugar de trabajar únicamente con direcciones IP.

A partir de la topología propuesta por la cátedra, se configura el direccionamiento IP, se habilita OSPF como protocolo de enrutamiento interno y luego se activa MPLS en las interfaces físicas. Sobre esta base se analizan las tablas de encaminamiento (FIB), las tablas de etiquetas (LIB y LFIB), el funcionamiento de los protocolos de distribución de etiquetas (TDP/LDP) y el efecto del ajuste de la MTU cuando se apilan etiquetas. Finalmente, se capturan tramas con un analizador de protocolos para identificar en qué momentos los paquetes ICMP viajan encapsulados en MPLS.

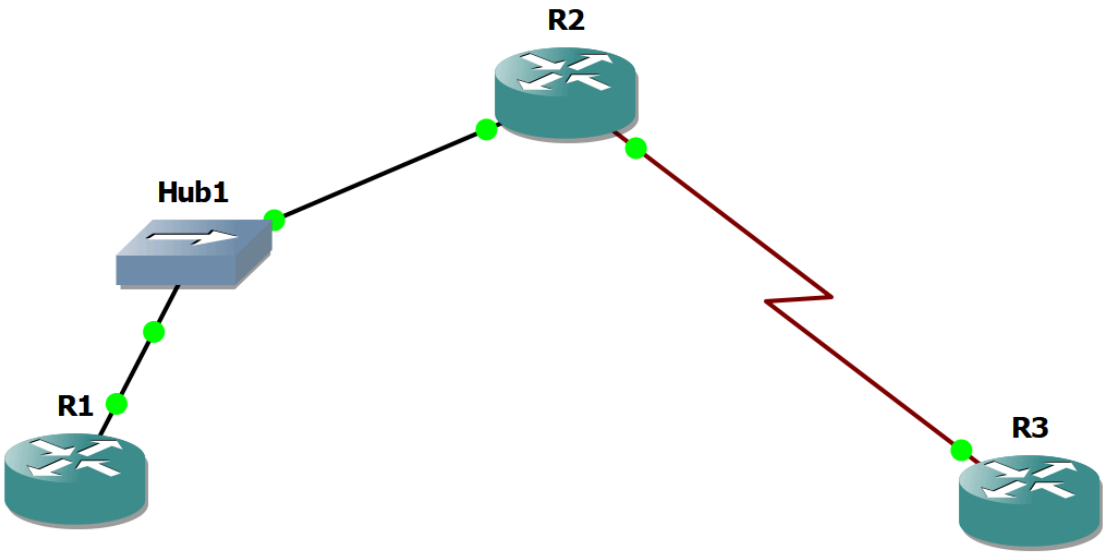
# 2 Descripción de la red MPLS

## 2.1 Topología lógica

La red utilizada es una maqueta simple formada por tres routers (R1, R2 y R3) conectados en serie.

- Entre R1 y R2 se implementa un enlace Ethernet en la red 172.16.12.0/24.
- Entre R2 y R3 se utiliza un enlace serie en la red 172.16.23.0/24.
- Cada router dispone además de una interfaz de loopback que representa una LAN interna: 172.16.1.0/24 en R1, 172.16.2.0/24 en R2 y 172.16.3.0/24 en R3.

Esta topología permite que R2 actúe como LSR intermedio dentro de la red MPLS, mientras que R1 y R3 funcionan como puntos de ingreso y egreso del tráfico etiquetado.




## 2.2 Esquema de direccionamiento

El direccionamiento IP respeta el planteo de la práctica:

- R1: Loopback0 172.16.1.1/24, Ethernet0/0 172.16.12.1/24

- R2: Loopback0 172.16.2.1/24, Ethernet0/0 172.16.12.2/24, Serial0/0 172.16.23.2/24
- R3: Loopback0 172.16.3.1/24, Serial0/0 172.16.23.3/24

Con esta asignación se consiguen redes claramente separadas para los enlaces punto a punto entre routers y para las LAN simuladas en las interfaces de loopback.

### 2.3 Herramientas utilizadas

Para la resolución de la práctica se utiliza el entorno GNS3 con routers Cisco virtualizados y Wireshark como analizador de protocolos. La imagen de IOS empleada corresponde a la familia 7200 con servicios avanzados, que incluye soporte para OSPF, CEF y MPLS.

En lugar de reproducir la maqueta desde cero, se trabaja a partir de capturas de consola y pantallas de GNS3 que muestran la configuración y el resultado de los comandos ejecutados sobre los routers.

# 3 Desarrollo de la práctica

## 3.1 Paso 0: Cableado e inicialización de los routers

En primer lugar se arma la maqueta física/lógica de la red MPLS propuesta en la guía.

La topología está formada por tres routers (R1, R2 y R3) conectados en serie.

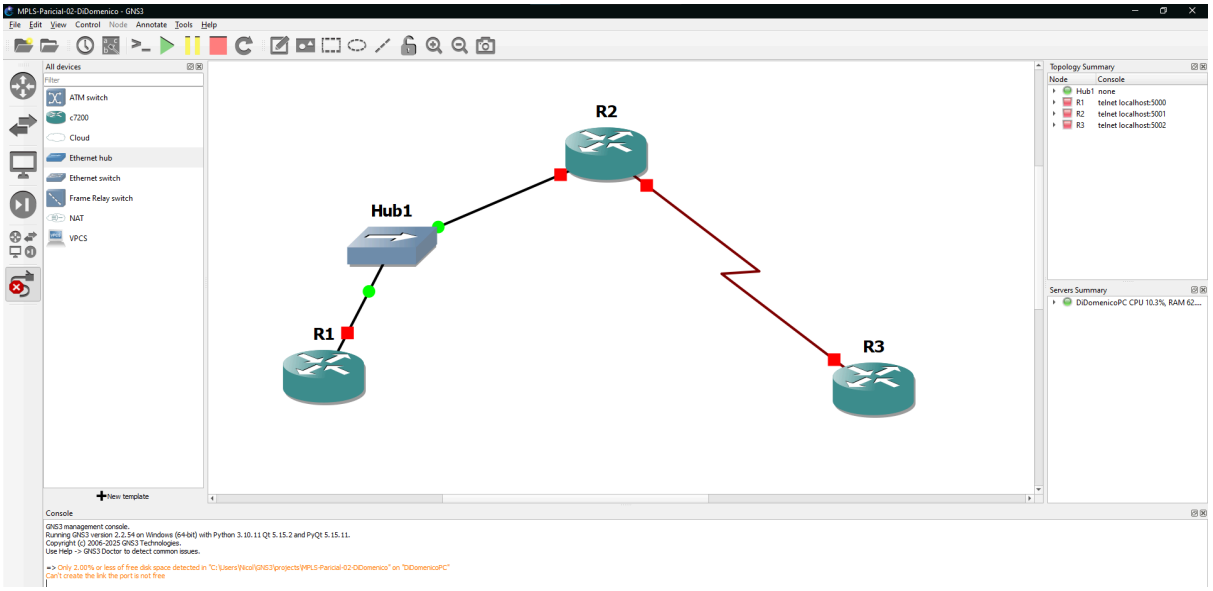
Entre R1 y R2 se reemplaza el switch del esquema original por un hub, con el fin de poder analizar el tráfico de la LAN con un sniffer externo (Wireshark).

R1 y R2 se conectan mediante un enlace Ethernet en la red 172.16.12.0/24, mientras que R2 y R3 se enlazan a través de una interfaz serie en la red 172.16.23.0/24.

Una vez realizado el cableado, se encienden los routers, se borra cualquier configuración previa si fuera necesario y se asignan los nombres de host básicos (R1, R2 y R3). Finalmente, se verifica que las interfaces requeridas estén en estado up/down (activas administrativamente pero aún sin dirección IP).

A continuación se muestran las capturas de la inicialización de cada router y de la verificación de interfaces:

***Topología de la maqueta en GNS3***



### *Inicialización de R1*

```
R1#enable
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#hostname R1
R1(config)#no ip domain-lookup
R1(config)#exit
R1#
*Nov 15 00:20:02.439: %SYS-5-CONFIG_I: Configured from console by console
R1#
```

### *Inicialización de R2*

```
R2#enable
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#hostname R2
R2(config)#no ip domain-lookup
R2(config)#exit
R2#
*Nov 15 00:24:18.063: %SYS-5-CONFIG_I: Configured from console by console
R2#
```

### *Inicialización de R3*

```
R3#enable
R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#hostname R3
R3(config)#no ip domain-lookup
R3(config)#exit
R3#
*Nov 15 00:26:11.643: %SYS-5-CONFIG_I: Configured from console by console
R3#
```

### *Verificación de interfaces con show ip interface brief en R1*

```
R1#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        unassigned      YES unset  administratively down down
Serial2/0              unassigned      YES unset  administratively down down
Serial2/1              unassigned      YES unset  administratively down down
Serial2/2              unassigned      YES unset  administratively down down
Serial2/3              unassigned      YES unset  administratively down down
```

### 3.2 Paso 1: Configuración del direccionamiento IP

A continuación se configura el direccionamiento IP en las interfaces físicas y de loopback de cada router, respetando el esquema definido en la práctica:

- **R1**
  - `Loopback0`: 172.16.1.1 /24
  - `FastEthernet0/0` (hacia R2): 172.16.12.1 /24
- **R2**
  - `Loopback0`: 172.16.2.1 /24
  - `FastEthernet0/0` (hacia R1): 172.16.12.2 /24
  - `Serial2/0` (hacia R3): 172.16.23.2 /24
- **R3**
  - `Loopback0`: 172.16.3.1 /24
  - `Serial2/0` (hacia R2): 172.16.23.3 /24

En las interfaces de loopback y FastEthernet solo se asigna la dirección IP y se habilitan con el comando no shutdown.

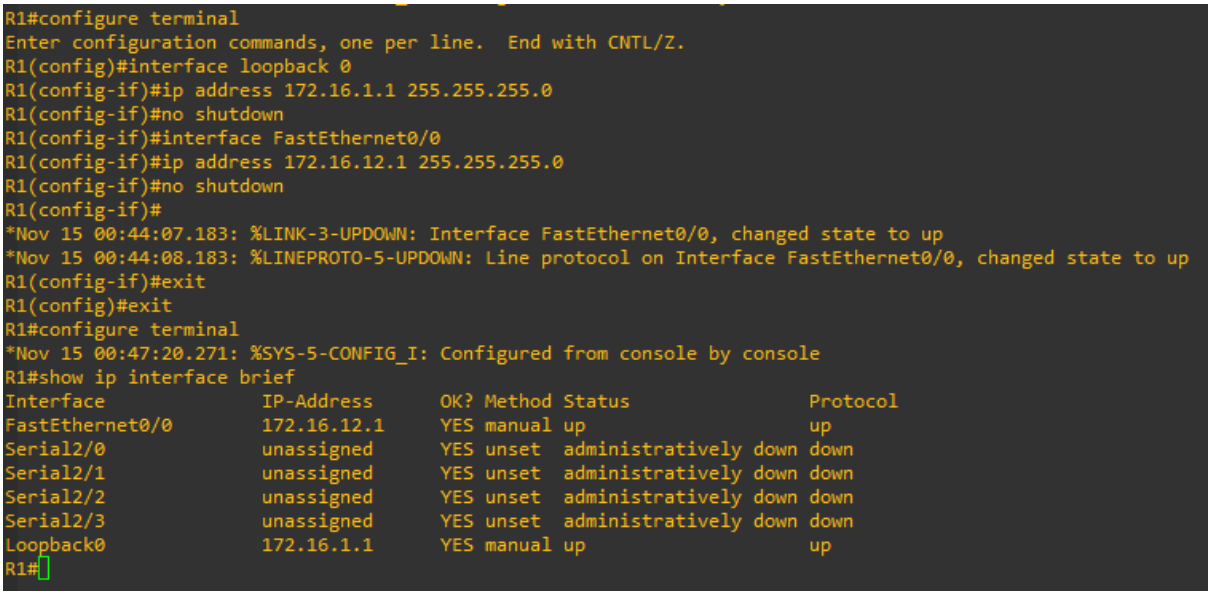
En el enlace serie entre R2 y R3, además de la IP se configura el clock rate en el extremo DCE para simular la temporización del enlace y luego se habilitan las interfaces.

Una vez configuradas todas las interfaces, en cada router se ejecuta el comando show ip interface brief para verificar que:

- cada interfaz tenga la dirección IP correcta, y
- el estado administrativo y el protocolo estén **up/up**.

A continuación se muestran las salidas de configuración y verificación:

***R1 – Configuración de Loopback0 y FastEthernet0/0, y salida de `show ip interface brief`***



```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface loopback 0
R1(config-if)#ip address 172.16.1.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#interface FastEthernet0/0
R1(config-if)#ip address 172.16.12.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#
*Nov 15 00:44:07.183: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Nov 15 00:44:08.183: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R1(config-if)#exit
R1(config)#exit
R1#configure terminal
*Nov 15 00:47:20.271: %SYS-5-CONFIG_I: Configured from console by console
R1#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        172.16.12.1     YES manual up                    up
Serial2/0              unassigned      YES unset  administratively down down
Serial2/1              unassigned      YES unset  administratively down down
Serial2/2              unassigned      YES unset  administratively down down
Serial2/3              unassigned      YES unset  administratively down down
Loopback0              172.16.1.1      YES manual up                    up
R1#
```

***R2 – Configuración de Loopback0, FastEthernet0/0 y Serial2/0, y primera verificación de `show ip interface brief`***

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface loopback 0
R2(config-if)#
*Nov 15 00:52:06.675: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R2(config-if)#ip address 172.16.2.1 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#interface FastEthernet0/0
R2(config-if)#ip address 172.16.12.2 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#
*Nov 15 00:55:24.923: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Nov 15 00:55:25.923: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R2(config-if)#interface serial 2/0
R2(config-if)#ip address 172.16.23.2 255.255.255.0
R2(config-if)#clockrate 64000
R2(config-if)#no shutdown
R2(config-if)#
*Nov 15 00:59:20.403: %LINK-3-UPDOWN: Interface Serial2/0, changed state to up
R2(config-if)#
*Nov 15 00:59:21.411: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
R2(config-if)#
*Nov 15 00:59:50.271: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
R2(config-if)#show ip interface brief
               ^
% Invalid input detected at '^' marker.

R2(config-if)#exit
R2(config)#exit
R2#
*Nov 15 01:03:38.627: %SYS-5-CONFIG_I: Configured from console by console
R2#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        172.16.12.2     YES manual up                    up
Serial2/0              172.16.23.2     YES manual up                    down
Serial2/1              unassigned      YES unset  administratively down down
Serial2/2              unassigned      YES unset  administratively down down
Serial2/3              unassigned      YES unset  administratively down down
Loopback0              172.16.2.1      YES manual up                    up
R2#
*Nov 15 01:18:00.271: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
R2#
```

***R3 – Configuración de Loopback0 y Serial2/0, y salida de `show ip interface brief`***

```
R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#interface loopback 0
R3(config-if)#
*Nov 15 01:14:21.715: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R3(config-if)#ip address 172.16.3.1 255.255.255.0
R3(config-if)#no shutdown
R3(config-if)#exit
R3(config)#interface Serial2/0
R3(config-if)#ip address 172.16.23.3 255.255.255.0
R3(config-if)#clock rate 64000
R3(config-if)#no shutdown
R3(config-if)#
*Nov 15 01:17:58.943: %LINK-3-UPDOWN: Interface Serial2/0, changed state to up
R3(config-if)#
*Nov 15 01:17:59.951: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
R3(config-if)#exit
R3(config)#exit
R3#
*Nov 15 01:18:11.847: %SYS-5-CONFIG_I: Configured from console by console
R3#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        unassigned      YES unset  administratively down down
Serial2/0              172.16.23.3     YES manual up                    up
Serial2/1              unassigned      YES unset  administratively down down
Serial2/2              unassigned      YES unset  administratively down down
Serial2/3              unassigned      YES unset  administratively down down
Loopback0              172.16.3.1      YES manual up                    up
R3#
```

Por último, se vuelve a ejecutar show ip interface brief en R2 para comprobar que el enlace serie con R3 quedó correctamente levantado (estado up/up tanto en Serial2/0 como en Loopback0 y FastEthernet0/0).

***R2 – Verificación final del estado de las interfaces***

```
R2#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        172.16.12.2     YES manual up                    up
Serial2/0              172.16.23.2     YES manual up                    up
Serial2/1              unassigned      YES unset  administratively down down
Serial2/2              unassigned      YES unset  administratively down down
Serial2/3              unassigned      YES unset  administratively down down
Loopback0              172.16.2.1      YES manual up                    up
R2#
```

### 3.3 Paso 2: Configurar OSPF en todos los routers

Sobre la red IP ya configurada se activa OSPF como protocolo de enrutamiento interno.

En cada router se define el proceso ospf 1 y se anuncia la red mayor 172.16.0.0/16

con máscara comodín 0.0.255.255 en el área 0, de modo que todas las subredes configuradas (enlaces y loopbacks) queden incluidas:

- router ospf 1
- network 172.16.0.0 0.0.255.255 area 0

Una vez aplicada la configuración, se comprueba la formación de vecindades OSPF con show ip ospf neighbor y se revisan las tablas de enrutamiento con show ip route.

En el caso de R1, por ejemplo, aparecen como rutas OSPF (código O) las redes de loopback de R2 y R3, lo que confirma que los routers intercambiaron correctamente la información de topología.

A continuación se muestran las salidas de la configuración de OSPF en cada router:

***R1 – Activación de OSPF proceso 1 y anuncio de la red 172.16.0.0/16***

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router ospf 1
R1(config-router)#network 172.16.0.0 0.0.255.255 area 0
R1(config-router)#exit
R1(config)#exit
R1#
*Nov 15 01:39:23.807: %SYS-5-CONFIG_I: Configured from console by console
R1#
```

***R2 – Configuración de OSPF y mensaje de formación de la vecindad con R1 (FastEthernet0/0)***

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#router ospf 1
R2(config-router)#network 172.16.0.0 0.0.255.255 area 0
R2(config-router)#exit
R2(config)#
*Nov 15 01:40:23.127: %OSPF-5-ADJCHG: Process 1, Nbr 172.16.1.1 on FastEthernet0/0 from LOADING to FULL, Loading Done
R2(config)#exit
R2#
*Nov 15 01:40:33.183: %SYS-5-CONFIG_I: Configured from console by console
R2#
```

***R3 – Configuración de OSPF y mensaje de formación de la vecindad con R2 (Serial2/0)***

```
R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#router ospf 1
R3(config-router)#network 172.16.0.0 0.0.255.255 area 0
R3(config-router)#
*Nov 15 01:42:48.711: %OSPF-5-ADJCHG: Process 1, Nbr 172.16.2.1 on Serial2/0 from LOADING to FULL, Loading Done
R3(config-router)#exit
R3(config)#exit
R3#
*Nov 15 01:42:59.135: %SYS-5-CONFIG_I: Configured from console by console
R3#
```

***¿Podría funcionar MPLS si no hubiera conectividad IP?***

No, ya que MPLS depende de que primero exista conectividad IP entre los routers. Los protocolos de distribución de etiquetas (TDP/LDP) utilizan mensajes IP y sesiones TCP para intercambiar etiquetas entre vecinos. Si no hay conectividad IP previa, no se forman las adyacencias de enrutamiento (por ejemplo OSPF), no se llenan las tablas de rutas y, por lo tanto, no se pueden definir las FEC ni establecer vecindades TDP/LDP.

Esto se ve en las capturas: recién después de tener las interfaces up y las rutas OSPF en show ip route aparecen los vecinos en show mpls ldp neighbor y se empiezan a anunciar etiquetas.

### 3.4 Paso 3: Comprobación de la conectividad IP y del funcionamiento de CEF (Cisco Express Forwarding)

Con OSPF operativo se realiza una verificación exhaustiva de conectividad IP.

Primero se revisan las tablas de enrutamiento con show ip route en los tres routers, donde se observa que todas las redes (enlaces de tránsito y loopbacks) aparecen aprendidas con código O (OSPF) o C (connected), sin rutas faltantes.

***R1 – Tabla de enrutamiento (`show ip route`)***

```
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 7 subnets, 2 masks
C        172.16.1.0/24 is directly connected, Loopback0
L        172.16.1.1/32 is directly connected, Loopback0
O        172.16.2.1/32 [110/2] via 172.16.12.2, 00:14:51, FastEthernet0/0
O        172.16.3.1/32 [110/66] via 172.16.12.2, 00:12:25, FastEthernet0/0
C        172.16.12.0/24 is directly connected, FastEthernet0/0
L        172.16.12.1/32 is directly connected, FastEthernet0/0
O        172.16.23.0/24 [110/65] via 172.16.12.2, 00:14:51, FastEthernet0/0
R1#
```

***R2 – Tabla de enrutamiento (`show ip route`)***

```
R2#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 8 subnets, 2 masks
O        172.16.1.1/32 [110/2] via 172.16.12.1, 00:14:51, FastEthernet0/0
C        172.16.2.0/24 is directly connected, Loopback0
L        172.16.2.1/32 is directly connected, Loopback0
O        172.16.3.1/32 [110/65] via 172.16.23.3, 00:12:28, Serial2/0
C        172.16.12.0/24 is directly connected, FastEthernet0/0
L        172.16.12.2/32 is directly connected, FastEthernet0/0
C        172.16.23.0/24 is directly connected, Serial2/0
L        172.16.23.2/32 is directly connected, Serial2/0
R2#
```

***R3 – Tabla de enrutamiento (`show ip route`)***

```
R3#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 7 subnets, 2 masks
O        172.16.1.1/32 [110/66] via 172.16.23.2, 00:12:33, Serial2/0
O        172.16.2.1/32 [110/65] via 172.16.23.2, 00:12:33, Serial2/0
C        172.16.3.0/24 is directly connected, Loopback0
L        172.16.3.1/32 is directly connected, Loopback0
O        172.16.12.0/24 [110/65] via 172.16.23.2, 00:12:33, Serial2/0
C        172.16.23.0/24 is directly connected, Serial2/0
L        172.16.23.3/32 is directly connected, Serial2/0
R3#
```

***Pruebas de ping***

A continuación se comprueba la conectividad IP enviando pings desde R1 hacia:

- La Loopback0 de R2: 172.16.2.1
- La Loopback0 de R3: 172.16.3.1
- La red de tránsito serie R2–R3: 172.16.23.3

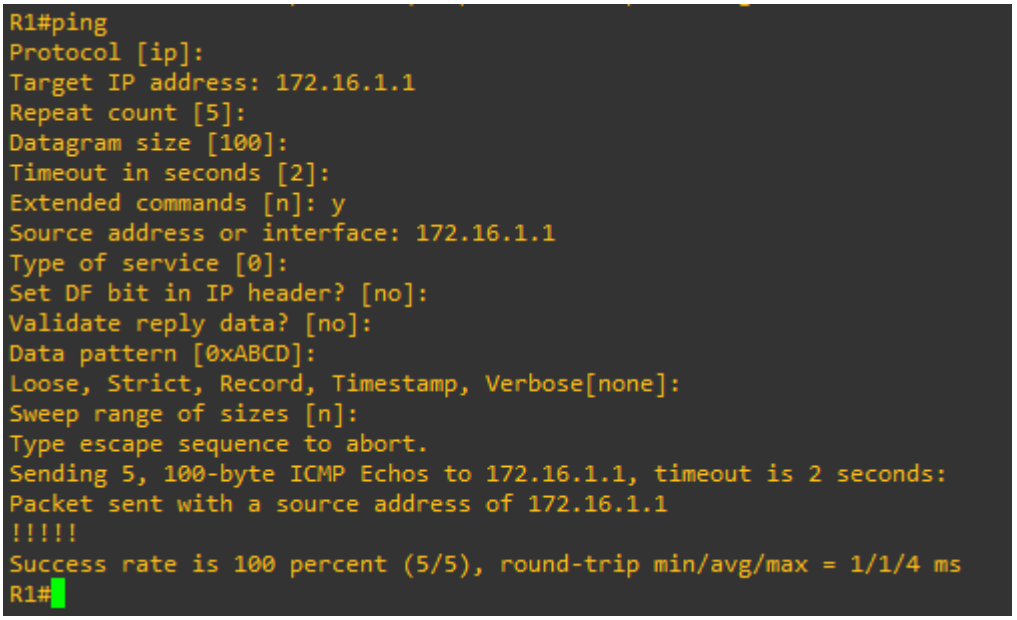
En todos los casos la tasa de éxito es del 100 %, lo que confirma que el enrutamiento IP funciona correctamente y que no hay errores de direccionamiento.

***Ping desde R1 a R2, R3 y a la red serie R2–R3***

```
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 7 subnets, 2 masks
C        172.16.1.0/24 is directly connected, Loopback0
L        172.16.1.1/32 is directly connected, Loopback0
O        172.16.2.1/32 [110/2] via 172.16.12.2, 00:14:51, FastEthernet0/0
O        172.16.3.1/32 [110/66] via 172.16.12.2, 00:12:25, FastEthernet0/0
C        172.16.12.0/24 is directly connected, FastEthernet0/0
L        172.16.12.1/32 is directly connected, FastEthernet0/0
O        172.16.23.0/24 [110/65] via 172.16.12.2, 00:14:51, FastEthernet0/0
R1#ping 172.16.2.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/13/28 ms
R1#172.16.3.1
   ^
% Invalid input detected at '^' marker.

R1#ping 172.16.3.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/32/32 ms
R1#ping 172.16.23.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.23.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/40/44 ms
R1#
```

Luego se realiza un ping extendido desde R1, utilizando como origen la dirección de la Loopback0 de R1 (172.16.1.1) para verificar que esa IP también tenga conectividad completa a través de la red.

***Ping extendido desde R1 con origen 172.16.1.1***




***¿Existe alguna interfaz que no conteste al ping?***

No. Todas las interfaces de R1, R2 y R3 responden correctamente a los pings enviados (tanto normales como extendidos), con una tasa de éxito del 100 %.

***Comprobación del camino con traceroute***

Para observar el camino que siguen los paquetes se ejecuta traceroute:

- Desde R1 hacia la Loopback0 de R3 (172.16.3.1). El resultado muestra que el tráfico pasa primero por R2 (salto 172.16.12.2) y luego por el enlace 172.16.23.3 hasta llegar a R3.
- Desde R3 hacia la Loopback0 de R1 (172.16.1.1). En sentido inverso, los paquetes pasan primero por R2 (salto 172.16.23.2) y luego por el enlace 172.16.12.1 hasta llegar a R1.

Esto confirma que el camino lógico que sigue el traceroute coincide con la información mostrada en las tablas de enrutamiento OSPF.

***Traceroute desde R1 a R3 (172.16.3.1)***

```
R1#traceroute 172.16.3.1
Type escape sequence to abort.
Tracing the route to 172.16.3.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.12.2 16 msec 20 msec 20 msec
  2 172.16.23.3 44 msec 44 msec 44 msec
R1#
```

***Traceroute desde R3 a R1 (172.16.1.1)***

```
R3#traceroute 172.16.1.1
Type escape sequence to abort.
Tracing the route to 172.16.1.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.23.2 16 msec 20 msec 20 msec
  2 172.16.12.1 44 msec 44 msec 44 msec
R3#
```

***¿Cuál es la función de CEF?***

CEF (Cisco Express Forwarding) se encarga de acelerar el reenvío de paquetes IP construyendo una tabla de reenvío (FIB) donde asocia cada prefijo de red con la interfaz de salida y, si corresponde, con la dirección IP del siguiente salto. Esta tabla se utiliza para reenviar los paquetes de forma rápida y eficiente, y sirve como base para que MPLS pueda trabajar con etiquetas sobre esas rutas IP.

Al ejecutar el comando show ip cef en R1 se observa cómo CEF asocia cada prefijo 172.16.x.x con su siguiente salto y la interfaz de salida correspondiente, lo que confirma que CEF está habilitado y funcionando correctamente.

***R1 – Tabla CEF (`show ip cef`)***

```
R1#show ip cef
Prefix              Next Hop            Interface
0.0.0.0/0           no route
0.0.0.0/8           drop
0.0.0.0/32          receive
127.0.0.0/8         drop
172.16.1.0/24       attached            Loopback0
172.16.1.0/32       receive             Loopback0
172.16.1.1/32       receive             Loopback0
172.16.1.255/32     receive             Loopback0
172.16.2.1/32       172.16.12.2         FastEthernet0/0
172.16.3.1/32       172.16.12.2         FastEthernet0/0
172.16.12.0/24      attached            FastEthernet0/0
172.16.12.0/32      receive             FastEthernet0/0
172.16.12.1/32      receive             FastEthernet0/0
172.16.12.2/32      attached            FastEthernet0/0
172.16.12.255/32    receive             FastEthernet0/0
172.16.23.0/24      172.16.12.2         FastEthernet0/0
224.0.0.0/4         drop
224.0.0.0/24        receive
240.0.0.0/4         drop
255.255.255.255/32  receive
R1#
```

### 3.5 Paso 4: Habilita MPLS en todas las interfaces físicas

Una vez validado el plano IP, se habilita MPLS en las interfaces físicas que interconectan los routers, tal como indica la guía:

- En R1, se activa MPLS sobre la interfaz FastEthernet0/0.
- En R2, se activa MPLS sobre FastEthernet0/0 y Serial2/0.
- En R3, se activa MPLS sobre Serial2/0.

Para ello se utiliza el comando mpls ip dentro de cada interfaz.

Al habilitar MPLS en ambos extremos de un enlace, el IOS muestra mensajes de syslog del tipo LDP Neighbor … is UP, que indican que se han creado nuevas vecindades TDP/LDP y que ha comenzado el intercambio de etiquetas entre los routers.

### *Configuración MPLS en R1*

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface FastEthernet0/0
R1(config-if)#mpls ip
R1(config-if)#exit
R1(config)#exit
R1#
*Nov 15 02:44:00.623: %SYS-5-CONFIG_I: Configured from console by console
R1#
*Nov 15 02:44:22.395: %LDP-5-NBRCHG: LDP Neighbor 172.16.2.1:0 (1) is UP
R1#
```

### *Configuración MPLS en R2*

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface FastEthernet0/0
R2(config-if)#mpls ip
R2(config-if)#
*Nov 15 02:44:22.363: %LDP-5-NBRCHG: LDP Neighbor 172.16.1.1:0 (1) is UP
R2(config-if)#exit
R2(config)#interface Serial2/0
R2(config-if)#mpls ip
R2(config-if)#exit
R2(config)#exit
R2#
*Nov 15 02:44:54.367: %SYS-5-CONFIG_I: Configured from console by console
R2#
*Nov 15 02:45:56.363: %LDP-5-NBRCHG: LDP Neighbor 172.16.3.1:0 (2) is UP
R2#
```

### *Configuración MPLS en R3*

```
R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#interface Serial2/0
R3(config-if)#mpls ip
R3(config-if)#
*Nov 15 02:45:56.307: %LDP-5-NBRCHG: LDP Neighbor 172.16.2.1:0 (1) is UP
R3(config-if)#exit
R3(config)#exit
R3#
*Nov 15 02:46:09.487: %SYS-5-CONFIG_I: Configured from console by console
R3#
```

### 3.6 Paso 5: Verifica la configuración de MPLS

Para comprobar que MPLS quedó correctamente habilitado se utilizan los comandos de diagnóstico disponibles:
show mpls interfaces, show mpls ldp discovery y show mpls ldp neighbor, entre otros.

El comando show mpls interfaces permite verificar, en cada router, qué interfaces tienen MPLS activo y qué protocolo de distribución de etiquetas se está utilizando (TDP o LDP). En la maqueta, todas las interfaces de tránsito aparecen con MPLS habilitado y con LDP en funcionamiento.

A continuación se revisa el funcionamiento de LDP mediante los comandos show mpls ldp discovery y show mpls ldp neighbor. En estos comandos se observa:

- El identificador local de LDP (LDP Identifier) de cada router.
- Las interfaces desde las que se descubren vecinos (Discovery Sources).
- La lista de vecinos LDP y el estado de las sesiones TCP establecidas con ellos.

Esto confirma que los routers se están descubriendo mutuamente y que están intercambiando bindings de etiquetas a través de conexiones TCP confiables.

***Comando `show mpls ?` en R1***

```
R1#show mpls ?
  discovery         Information about LSP discovery
  flow              MPLS netflow information
  forwarding-table  Show the Label Forwarding Table
  interfaces        Per-interface MPLS forwarding information
  ip                MPLS IP information
  l2transport       MPLS circuit transport info
  label             Label information
  ldp               Label Distribution Protocol information
  memory            Memory usage information
  mldp              Show MLDP routing protocol parameters
  oam               OAM information
  static            Show MPLS static information
  traffic-eng       Traffic engineering information

R1#show mpls
```

***Interfaces MPLS en R1***

```
R1#show mpls interfaces
Interface              IP           Tunnel   BGP Static Operational
FastEthernet0/0        Yes (ldp)    No       No  No     Yes
R1#
```

***Interfaces MPLS en R2***

```
R2#show mpls interfaces
Interface              IP           Tunnel   BGP Static Operational
FastEthernet0/0        Yes (ldp)    No       No  No     Yes
Serial2/0              Yes (ldp)    No       No  No     Yes
R2#
```

Interfaces MPLS en R3

```
R3#show mpls interfaces
Interface              IP           Tunnel   BGP Static Operational
Serial2/0              Yes (ldp)    No       No  No     Yes
R3#
```

***Comando `show mpls ldp discovery` en R1***

```
R1#show mpls ldp discovery
 Local LDP Identifier:
    172.16.1.1:0
    Discovery Sources:
    Interfaces:
        FastEthernet0/0 (ldp): xmit/recv
            LDP Id: 172.16.2.1:0
R1#
```

***Comando `show mpls ldp neighbor` en R1***

```
R1#show mpls ldp nei
    Peer LDP Ident: 172.16.2.1:0; Local LDP Ident 172.16.1.1:0
        TCP connection: 172.16.2.1.29050 - 172.16.1.1.646
        State: Oper; Msgs sent/rcvd: 31/31; Downstream
        Up time: 00:20:27
        LDP discovery sources:
          FastEthernet0/0, Src IP addr: 172.16.12.2
        Addresses bound to peer LDP Ident:
          172.16.12.2     172.16.23.2     172.16.2.1
R1#
```

***Comando `show mpls ldp discovery` en R2***

```
R2#show mpls ldp discovery
 Local LDP Identifier:
    172.16.2.1:0
    Discovery Sources:
    Interfaces:
        FastEthernet0/0 (ldp): xmit/recv
            LDP Id: 172.16.1.1:0
        Serial2/0 (ldp): xmit/recv
            LDP Id: 172.16.3.1:0
R2#
```

***Comando `show mpls ldp neighbor` en R2***

```
R2#show mpls ldp nei
    Peer LDP Ident: 172.16.1.1:0; Local LDP Ident 172.16.2.1:0
        TCP connection: 172.16.1.1.646 - 172.16.2.1.29050
        State: Oper; Msgs sent/rcvd: 34/34; Downstream
        Up time: 00:23:10
        LDP discovery sources:
          FastEthernet0/0, Src IP addr: 172.16.12.1
        Addresses bound to peer LDP Ident:
          172.16.12.1     172.16.1.1
    Peer LDP Ident: 172.16.3.1:0; Local LDP Ident 172.16.2.1:0
        TCP connection: 172.16.3.1.42573 - 172.16.2.1.646
        State: Oper; Msgs sent/rcvd: 33/32; Downstream
        Up time: 00:21:36
        LDP discovery sources:
          Serial2/0, Src IP addr: 172.16.23.3
        Addresses bound to peer LDP Ident:
          172.16.23.3     172.16.3.1
R2#
```

***Comando `show mpls ldp discovery` en R3***

```
R3#show mpls ldp discovery
 Local LDP Identifier:
    172.16.3.1:0
    Discovery Sources:
    Interfaces:
        Serial2/0 (ldp): xmit/recv
            LDP Id: 172.16.2.1:0
R3#
```

***Comando `show mpls ldp neighbor` en R3***

```
R3#show mpls ldp nei
    Peer LDP Ident: 172.16.2.1:0; Local LDP Ident 172.16.3.1:0
        TCP connection: 172.16.2.1.646 - 172.16.3.1.42573
        State: Oper; Msgs sent/rcvd: 35/36; Downstream
        Up time: 00:24:17
        LDP discovery sources:
          Serial2/0, Src IP addr: 172.16.23.2
        Addresses bound to peer LDP Ident:
          172.16.12.2     172.16.23.2     172.16.2.1
R3#
```

***¿Qué protocolo de transporte utiliza LDP (o TDP) para comunicarse con sus vecinos?***

LDP (Label Distribution Protocol) y el antiguo TDP (Tag Distribution Protocol) utilizan TCP como protocolo de transporte. Al apoyarse en TCP, la sesión de señalización entre routers MPLS es confiable y orientada a la conexión, de modo que los bindings de etiquetas se intercambian sin pérdidas entre los LSR vecinos.

Esto se puede comprobar en la salida del comando show mpls ldp neighbor, donde aparece una línea como:

*"TCP connection: 172.16.2.1.29050 – 172.16.1.1.646"*

Allí se ve que la sesión LDP/TDP se establece sobre TCP (puerto 646 en el ejemplo).

En la siguiente captura también se observa que el estado de la sesión es Oper, lo que indica que la vecindad LDP está activa y que MPLS puede funcionar correctamente sobre ese enlace.

```
R1#show mpls ldp neighbor
    Peer LDP Ident: 172.16.2.1:0; Local LDP Ident 172.16.1.1:0
        TCP connection: 172.16.2.1.29050 - 172.16.1.1.646
        State: Oper; Msgs sent/rcvd: 41/42; Downstream
        Up time: 00:29:33
        LDP discovery sources:
          FastEthernet0/0, Src IP addr: 172.16.12.2
        Addresses bound to peer LDP Ident:
          172.16.12.2     172.16.23.2     172.16.2.1
R1#
```

### 3.7 Paso 6: Estudio de las tablas LIB y LFIB

El siguiente paso consiste en estudiar cómo MPLS traduce las rutas IP a etiquetas.

Para ello se utilizan los comandos show mpls ldp bindings (LIB) y show mpls forwarding-table (LFIB).

La LIB (Label Information Base) muestra, para cada prefijo, las asociaciones de etiquetas locales (local binding) y las que anuncian los vecinos (remote binding). Esto permite identificar qué etiqueta asigna cada router para representar un determinado destino y qué etiquetas recibe de cada vecino para esa misma FEC.

La LFIB (Label Forwarding Information Base), en cambio, es la tabla efectiva de reenvío basada en etiquetas. Allí se observa, para cada etiqueta local, qué prefijo representa, por qué interfaz se reenvía el tráfico, cuál es el siguiente salto y qué etiqueta de salida (si corresponde) se utiliza. En el caso de R2, por ejemplo, se aprecia cómo el router tiene una etiqueta local para la loopback de R1 y otra para la de R3, y en algunos casos la etiqueta de salida figura como No Label, indicando que se realiza Penultimate Hop Popping (PHP) y se entrega un paquete IP puro al siguiente salto.

***Salida de `show mpls ldp bindings` en R1***

```
R1#show mpls ldp bindings
  lib entry: 172.16.1.0/24, rev 2
        local binding:  label: imp-null
  lib entry: 172.16.1.1/32, rev 11
        remote binding: lsr: 172.16.2.1:0, label: 16
  lib entry: 172.16.2.0/24, rev 12
        remote binding: lsr: 172.16.2.1:0, label: imp-null
  lib entry: 172.16.2.1/32, rev 4
        local binding:  label: 16
  lib entry: 172.16.3.1/32, rev 6
        local binding:  label: 17
        remote binding: lsr: 172.16.2.1:0, label: 17
  lib entry: 172.16.12.0/24, rev 8
        local binding:  label: imp-null
        remote binding: lsr: 172.16.2.1:0, label: imp-null
  lib entry: 172.16.23.0/24, rev 10
        local binding:  label: 18
        remote binding: lsr: 172.16.2.1:0, label: imp-null
R1#
```

***Salida de `show mpls ldp bindings` en R2***

```
R2#show mpls ldp bindings
  lib entry: 172.16.1.0/24, rev 11
        remote binding: lsr: 172.16.1.1:0, label: imp-null
  lib entry: 172.16.1.1/32, rev 2
        local binding:  label: 16
        remote binding: lsr: 172.16.3.1:0, label: 16
  lib entry: 172.16.2.0/24, rev 4
        local binding:  label: imp-null
  lib entry: 172.16.2.1/32, rev 12
        remote binding: lsr: 172.16.1.1:0, label: 16
        remote binding: lsr: 172.16.3.1:0, label: 17
  lib entry: 172.16.3.0/24, rev 13
        remote binding: lsr: 172.16.3.1:0, label: imp-null
  lib entry: 172.16.3.1/32, rev 6
        local binding:  label: 17
        remote binding: lsr: 172.16.1.1:0, label: 17
  lib entry: 172.16.12.0/24, rev 8
        local binding:  label: imp-null
        remote binding: lsr: 172.16.1.1:0, label: imp-null
        remote binding: lsr: 172.16.3.1:0, label: 18
  lib entry: 172.16.23.0/24, rev 10
        local binding:  label: imp-null
        remote binding: lsr: 172.16.1.1:0, label: 18
 --More--
```

***Salida de `show mpls ldp bindings` en R3***

```
R3#show mpls ldp bindings
  lib entry: 172.16.1.1/32, rev 2
        local binding:  label: 16
        remote binding: lsr: 172.16.2.1:0, label: 16
  lib entry: 172.16.2.0/24, rev 11
        remote binding: lsr: 172.16.2.1:0, label: imp-null
  lib entry: 172.16.2.1/32, rev 4
        local binding:  label: 17
  lib entry: 172.16.3.0/24, rev 6
        local binding:  label: imp-null
  lib entry: 172.16.3.1/32, rev 12
        remote binding: lsr: 172.16.2.1:0, label: 17
  lib entry: 172.16.12.0/24, rev 8
        local binding:  label: 18
        remote binding: lsr: 172.16.2.1:0, label: imp-null
  lib entry: 172.16.23.0/24, rev 10
        local binding:  label: imp-null
        remote binding: lsr: 172.16.2.1:0, label: imp-null
R3#
```

***¿Por qué para un mismo destino tiene varias etiquetas? ¿Con qué vecinos intercambia etiquetas?***

Un router puede tener varias etiquetas para un mismo destino porque cada vecino MPLS anuncia su propia etiqueta local (remote binding) para esa FEC.

Por ejemplo, R2 tiene dos vecinos (R1 y R3) y cada uno anuncia su etiqueta para alcanzar un destino determinado. R2 almacena todas esas etiquetas en su tabla LIB, pero solo utilizará, en la LFIB, la etiqueta del vecino que aparece como next-hop en la tabla de rutas IP.

Los routers intercambian etiquetas únicamente con sus vecinos MPLS directos, es decir, aquellos con los que comparten una interfaz donde está configurado mpls ip. En esta maqueta:

- R1 intercambia etiquetas solo con R2 (FastEthernet0/0).
- R2 intercambia etiquetas con R1 y R3 (FastEthernet0/0 y Serial2/0).
- R3 intercambia etiquetas solo con R2 (Serial2/0).

Esto se comprueba ejecutando show mpls ldp neighbor en cada router.

***Salida de `show mpls forwarding-table` en R1***

```
R1#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         No Label   172.16.2.1/32    0             Fa0/0      172.16.12.2
17         17         172.16.3.1/32    0             Fa0/0      172.16.12.2
18         Pop Label  172.16.23.0/24   0             Fa0/0      172.16.12.2
R1#
```

***Salida de `show mpls forwarding-table` en R2***

```
R2#sh mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         No Label   172.16.1.1/32    0             Fa0/0      172.16.12.1
17         No Label   172.16.3.1/32    0             Se2/0      point2point
R2#
```

***Salida de `show mpls forwarding-table` en R3***

```
R3#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         16         172.16.1.1/32    0             Se2/0      point2point
17         No Label   172.16.2.1/32    0             Se2/0      point2point
18         Pop Label  172.16.12.0/24   0             Se2/0      point2point
R3#
```

***Tablas resumen de la LFIB***

**R1**

| Network | LSR (next-hop) | Label / Acción |
|---|---|---|
| 172.16.2.1/32 | R2 (172.16.12.2) | No label |
| 172.16.3.1/32 | R2 (172.16.12.2) | 17 |
| 172.16.23.0/24 | R2 (172.16.12.2) | Pop label |

**R2**

| Network | LSR (next-hop) | Label / Acción |
|---|---|---|
| 172.16.1.1/32 | R1 (172.16.12.1) | No label |
| 172.16.3.1/32 | R3 (172.16.23.3) | No label |

**R3**

| Network | LSR (next-hop) | Label / Acción |
|---|---|---|
| 172.16.1.1/32 | R2 (172.16.23.2) | 16 |
| 172.16.2.1/32 | R2 (172.16.23.2) | No label |
| 172.16.12.0/24 | R2 (172.16.23.2) | Pop label |

***A la vista de los resultados mostrados en las tablas LIB y LFIB, si realizamos un ping desde R1 con origen en 172.16.1.1 y destino a 172.16.3.1 ¿qué etiquetas se utilizan?***

En la LFIB de R1 la FEC 172.16.3.1/32 sale por Fa0/0 hacia R2 con etiqueta 17.

En la LFIB de R2 la FEC 172.16.3.1/32 sale por Serial2/0 hacia R3 con "No Label" (porque R3 anunció implicit-NULL).

Entonces desde R1 a 172.16.3.1 se impone la etiqueta 17 en el enlace R1–R2, y en el enlace R2–R3 el paquete va sin etiqueta (R2 hace PHP y quita la etiqueta antes de enviarlo a R3).

***¿Qué significado tiene la entrada "local binding"?***

Local binding es la asociación entre una FEC (prefijo de la tabla de rutas) y la etiqueta que ese router asigna localmente para dicha FEC.

Es la etiqueta propia del router, la que él anuncia a sus vecinos LDP/TDP para que la usen cuando le envíen tráfico hacia ese destino.

***¿Qué significado tiene la entrada "remote binding"?***

Remote binding es la asociación entre una FEC y la etiqueta que anuncia un vecino LDP/TDP para esa misma FEC.

Es la etiqueta que el router debe utilizar cuando envía tráfico hacia ese vecino para alcanzar ese destino.

***En el router R2, ¿por qué hay más de una asociación remota para cada red?***

Porque R2 tiene dos vecinos MPLS (R1 y R3), y cada vecino anuncia su propia etiqueta para la misma FEC.

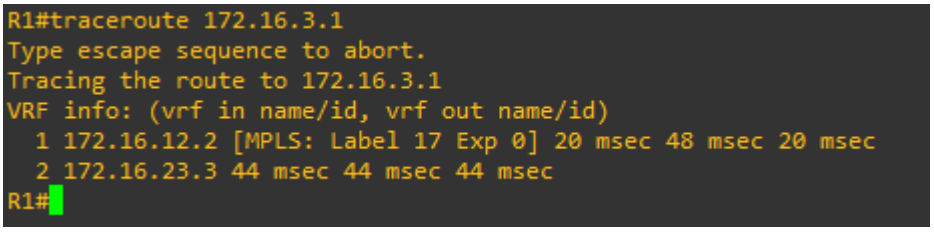
Por eso, en la LIB de R2 aparecen varias entradas remote binding para un mismo destino: una etiqueta remota anunciada por R1 y otra anunciada por R3.

***¿Qué significa la etiqueta "implicit NULL"?***

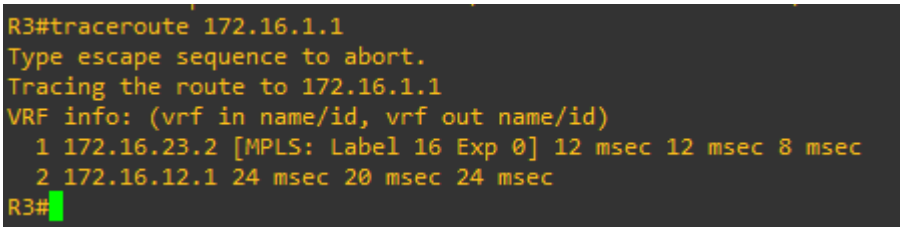
La etiqueta implicit-NULL (valor 3) indica al router anterior que debe eliminar la etiqueta (pop) antes de enviar el paquete al router destino.

Se usa para implementar PHP (Penultimate Hop Popping): el penúltimo salto quita la etiqueta y envía el paquete como IP “pelado”, de modo que el router de salida (egress LER) no tenga que hacer conmutación MPLS, solo un reenvío IP normal.

***Traceroute con MPLS desde R1 hacia la Loopback de R3 (172.16.3.1)***




***Traceroute con MPLS desde R3 hacia la Loopback de R1 (172.16.1.1)***




***¿Qué diferencias observas y por qué?***

En el traceroute desde R1 a 172.16.3.1, en el primer salto (172.16.12.2, R2) aparece la información MPLS:

[MPLS: Label 17 Exp 0]

Esto indica que R1 envía el tráfico hacia R3 usando la etiqueta 17.

El segundo salto (172.16.23.3, R3) ya no muestra etiqueta porque R2 actúa como penultimate hop y realiza PHP, quitando la etiqueta antes de entregar el paquete al router de salida.

En el traceroute desde R3 a 172.16.1.1 sucede lo mismo en sentido inverso: en el primer salto (172.16.23.2, R2) aparece:

*“[MPLS: Label 16 Exp 0]”*

Es decir, R3 envía el tráfico hacia R1 con la etiqueta 16, y el segundo salto (172.16.12.1, R1) recibe el paquete sin etiqueta.

Por lo tanto, a diferencia del traceroute sin MPLS, ahora se ve una etiqueta MPLS solo en el primer salto de cada trayecto, y desaparece en el último salto debido al uso de PHP (implicit-NULL).

### 3.8 Paso 7: Migrando de TDP a LDP

Una vez verificado el funcionamiento de MPLS con TDP, se realiza la migración a LDP, tal como propone la práctica.

Primero se modifica la configuración de R2 para que utilice LDP como protocolo de distribución de etiquetas, mientras que R1 y R3 permanecen momentáneamente con TDP. Esta situación intermedia permite observar que las vecindades MPLS se pierden, ya que los protocolos no son compatibles entre sí: aunque el enrutamiento IP sigue funcionando, ya no se intercambian etiquetas entre los routers.

Posteriormente se homogeneiza la configuración en todo el dominio MPLS: R1 y R3 también se ajustan para utilizar LDP, y se verifica nuevamente con show mpls interfaces y show mpls ldp neighbor que las adyacencias vuelven a estar operativas, ahora bajo un único protocolo de distribución de etiquetas. De esta forma se completa la migración de TDP a LDP sin afectar la conectividad IP de la red.

***Configuración de R2 para utilizar LDP como protocolo de distribución de etiquetas***

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#mpls label protocol ldp
R2(config)#end
R2#
*Nov 15 03:54:34.491: %SYS-5-CONFIG_I: Configured from console by console
R2#
```

***Verificación de interfaces MPLS y vecinos LDP en R2 tras la migración de TDP a LDP***

```
R2#show mpls interfaces
Interface              IP            Tunnel   BGP Static Operational
FastEthernet0/0        Yes (ldp)     No       No  No     Yes
Serial2/0              Yes (ldp)     No       No  No     Yes
R2#show mpls ldp neighbor
    Peer LDP Ident: 172.16.1.1:0; Local LDP Ident 172.16.2.1:0
        TCP connection: 172.16.1.1.646 - 172.16.2.1.29050
        State: Oper; Msgs sent/rcvd: 92/90; Downstream
        Up time: 01:12:57
        LDP discovery sources:
          FastEthernet0/0, Src IP addr: 172.16.12.1
        Addresses bound to peer LDP Ident:
          172.16.12.1     172.16.1.1
    Peer LDP Ident: 172.16.3.1:0; Local LDP Ident 172.16.2.1:0
        TCP connection: 172.16.3.1.42573 - 172.16.2.1.646
        State: Oper; Msgs sent/rcvd: 89/88; Downstream
        Up time: 01:11:23
        LDP discovery sources:
          Serial2/0, Src IP addr: 172.16.23.3
        Addresses bound to peer LDP Ident:
          172.16.23.3     172.16.3.1
R2#show mpls ldp neighbor
    Peer LDP Ident: 172.16.1.1:0; Local LDP Ident 172.16.2.1:0
        TCP connection: 172.16.1.1.646 - 172.16.2.1.29050
        State: Oper; Msgs sent/rcvd: 92/90; Downstream
        Up time: 01:13:02
        LDP discovery sources:
          FastEthernet0/0, Src IP addr: 172.16.12.1
        Addresses bound to peer LDP Ident:
          172.16.12.1     172.16.1.1
    Peer LDP Ident: 172.16.3.1:0; Local LDP Ident 172.16.2.1:0
        TCP connection: 172.16.3.1.42573 - 172.16.2.1.646
        State: Oper; Msgs sent/rcvd: 90/89; Downstream
        Up time: 01:11:28
        LDP discovery sources:
          Serial2/0, Src IP addr: 172.16.23.3
        Addresses bound to peer LDP Ident:
          172.16.23.3     172.16.3.1
R2#show mpls ldp neighbor
    Peer LDP Ident: 172.16.1.1:0; Local LDP Ident 172.16.2.1:0
        TCP connection: 172.16.1.1.646 - 172.16.2.1.29050
        State: Oper; Msgs sent/rcvd: 92/90; Downstream
        Up time: 01:13:08
        LDP discovery sources:
          FastEthernet0/0, Src IP addr: 172.16.12.1
        Addresses bound to peer LDP Ident:
          172.16.12.1     172.16.1.1
    Peer LDP Ident: 172.16.3.1:0; Local LDP Ident 172.16.2.1:0
        TCP connection: 172.16.3.1.42573 - 172.16.2.1.646
        State: Oper; Msgs sent/rcvd: 90/89; Downstream
        Up time: 01:11:34
        LDP discovery sources:
          Serial2/0, Src IP addr: 172.16.23.3
        Addresses bound to peer LDP Ident:
          172.16.23.3     172.16.3.1
```

***Traceroute con MPLS entre R1 y R3 después de cambiar a LDP***

```
R1#traceroute 172.16.3.1
Type escape sequence to abort.
Tracing the route to 172.16.3.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.12.2 [MPLS: Label 17 Exp 0] 28 msec 20 msec 8 msec
  2 172.16.23.3 24 msec 20 msec 24 msec
R1#
```

```
R3#traceroute 172.16.1.1
Type escape sequence to abort.
Tracing the route to 172.16.1.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.23.2 [MPLS: Label 16 Exp 0] 32 msec 12 msec 20 msec
  2 172.16.12.1 32 msec 32 msec 32 msec
R3#
```

***Utilizando los comandos anteriores, intenta averiguar que ha pasado. Utiliza el ping, el traceroute y los comandos show vistos anteriormente. ¿Podrías describir qué ha pasado?***

Después de ejecutar el comando mpls label protocol ldp en R2, este router pasa a utilizar LDP como protocolo de distribución de etiquetas, mientras que R1 y R3 continúan trabajando con TDP.

Como ambos protocolos no son compatibles entre sí, R2 deja de establecer sesiones de distribución de etiquetas con sus vecinos. Esto se observa en los comandos show mpls ldp neighbor y show mpls ldp discovery, donde ya no aparecen (o aparecen caídas) las vecindades con R1 y R3.

A nivel de conectividad IP, los pings y traceroute siguen funcionando porque OSPF continúa intercambiando rutas correctamente. Sin embargo, ya no se ven etiquetas MPLS en los traceroute, y las tablas LIB/LFIB dejan de mostrar asociaciones remotas útiles. En resumen, MPLS deja de funcionar de forma adecuada en la red debido a un mismatch de protocolo de señalización (TDP vs LDP).

***¿Cómo podríamos solucionar el problema?***

Para resolver el problema, todos los routers del dominio MPLS deben utilizar el mismo protocolo de distribución de etiquetas.

La solución consiste en configurar también en R1 y R3 el uso de LDP:

- R1(config)#mpls label protocol ldp
- R3(config)#mpls label protocol ldp

Una vez aplicado el cambio, las sesiones LDP se restablecen (show mpls ldp neighbor vuelve a mostrar a R1–R2 y R2–R3 en estado Oper), las tablas LIB y LFIB vuelven a tener etiquetas remotas y en los traceroute se vuelven a observar labels MPLS en el primer salto. De esta forma se recupera el correcto funcionamiento de MPLS en toda la red.

Alternativamente, podríamos eliminar el comando en R2 (no mpls label protocol ldp) para que vuelva a TDP y coincida con R1 y R3, pero la migración recomendada es unificar todo en LDP.

### 3.9 Paso 8: Modifica el tamaño de MTU para MPLS

En este paso se analiza el impacto de MPLS en el tamaño máximo de las tramas.

La cabecera MPLS agrega 4 bytes por etiqueta, por lo que cuando se aplican varias etiquetas es necesario ajustar la MTU MPLS para evitar problemas de fragmentación o descarte de tramas.

Primero se comprueba la MTU actual en las interfaces que conectan R1 y R2, utilizando el comando show mpls interfaces fastethernet0/0 detail.

Inicialmente, la MTU reportada es de 1500 bytes, que coincide con el valor típico de una interfaz Ethernet.

***MTU MPLS en Fa0/0 antes del cambio – R1 y R2***

```
R1#show mpls interfaces fastethernet0/0 detail
Interface FastEthernet0/0:
        Type Unknown
        IP labeling enabled (ldp) :
          Interface config
        LSP Tunnel labeling not enabled
        IP FRR labeling not enabled
        BGP labeling not enabled
        MPLS operational
        MTU = 1500
R1#
```

```
R2#show mpls interfaces fastethernet0/0 detail
Interface FastEthernet0/0:
        Type Unknown
        IP labeling enabled (ldp) :
          Interface config
        LSP Tunnel labeling not enabled
        IP FRR labeling not enabled
        BGP labeling not enabled
        MPLS operational
        MTU = 1500
R2#
```

A continuación, siguiendo la consigna de la práctica, se intenta modificar la MTU específica de MPLS a 1508 bytes en las interfaces FastEthernet0/0 de R1 y R2, utilizando el comando mpls mtu 1508 dentro de la interfaz.

***Intento de cambio de MTU MPLS a 1508 bytes en R1***

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface FastEthernet0/0
R1(config-if)#mpls mtu 1508
                       ^
% Invalid input detected at '^' marker.

R1(config-if)#end
R1#
*Nov 15 04:08:07.135: %SYS-5-CONFIG_I: Configured from console by console
R1#
```

***Intento de cambio de MTU MPLS a 1508 bytes en R2***

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface FastEthernet0/0
R2(config-if)#mpls mtu 1508
                       ^
% Invalid input detected at '^' marker.

R2(config-if)#mpls mtu 1508
                       ^
% Invalid input detected at '^' marker.

R2(config-if)#
```

Con el comando show mpls interfaces fastethernet0/0 detail en R1 y R2 se comprueba nuevamente que la MTU utilizada por MPLS en el enlace FastEthernet0/0 es de 1500 bytes, es decir, el valor no cambia pese al intento de configuración.

Se intentó aumentar la MTU MPLS a 1508 bytes con el comando mpls mtu 1508 en la interfaz FastEthernet0/0 de R1 y R2, tal como indica la guía.
Sin embargo, el IOS utilizado en GNS3 devuelve el mensaje “Invalid input detected”, lo que indica que esta versión no soporta el subcomando mpls mtu.

Por este motivo, la MTU MPLS se mantiene en 1500 bytes, aunque teóricamente debería poder configurarse en 1508 para soportar hasta dos cabeceras MPLS adicionales sin superar el tamaño máximo de trama.

En esta maqueta, como solo se está utilizando una etiqueta MPLS por trayecto, el valor estándar de 1500 bytes resulta suficiente y no se observan problemas de fragmentación ni descartes de paquetes.

### 3.10 Paso 9: Análisis de tramas MPLS

Finalmente se analiza el tráfico MPLS a nivel de trama utilizando un analizador de protocolos. Para ello se conecta Wireshark en el segmento compartido por el hub entre R1 y R2, configurando la captura para registrar todo el tráfico de capa 2.

Con la captura en marcha, se generan pings desde la loopback de R1 (172.16.1.1) hacia la loopback de R3 (172.16.3.1). En Wireshark se filtra el tráfico relevante y se observa que los ICMP Echo Request (paquetes de ida de R1 hacia R3) salen encapsulados en MPLS, con la etiqueta asignada previamente en la

LIB/LFIB. En cambio, debido al mecanismo de Penultimate Hop Popping, la etiqueta se elimina en R2 antes de llegar a R3, por lo que las respuestas ICMP Echo Reply suelen regresar a R1 como paquetes IP puros sin etiqueta MPLS visible.

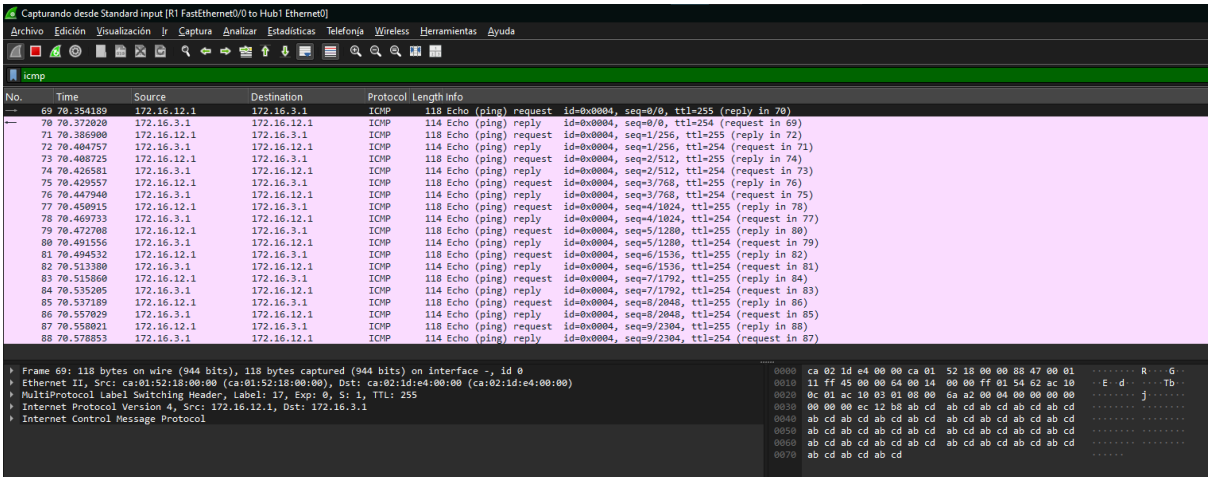
Este análisis permite relacionar directamente la teoría de MPLS con la práctica, identificando en las capturas cómo cambian los encabezados a medida que los paquetes atraviesan la red y en qué tramo del recorrido se aplica y se elimina la etiqueta.

**Ping desde R1 (172.16.12.1) a la Loopback de R3 (172.16.3.1)**

```
R1#ping 172.16.3.1 repeat 10
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 172.16.3.1, timeout is 2 seconds:
!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 20/25/56 ms
R1#
```

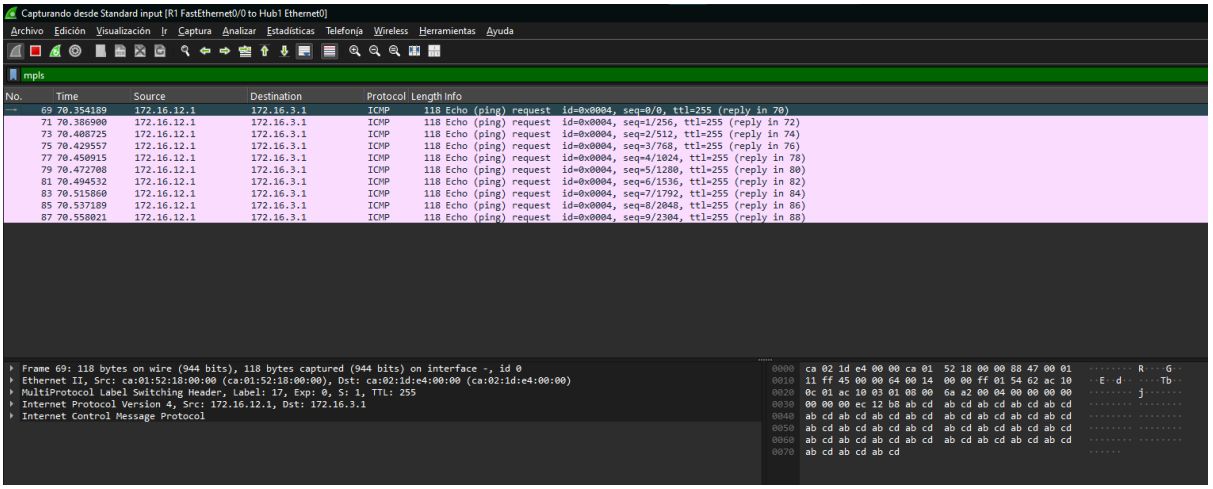
*Se verifica que el ping desde R1 hacia la dirección 172.16.3.1 tiene una tasa de éxito del 100 % (10/10), por lo que existe conectividad extremo a extremo entre R1 y R3 a través de la red MPLS.*

***Tráfico ICMP entre R1 y R3 (filtro `icmp`)***




*Con el filtro `icmp` se observan tanto los paquetes **ICMP Echo Request** enviados desde 172.16.12.1 a 172.16.3.1, como los **ICMP Echo Reply** de vuelta desde 172.16.3.1 hacia 172.16.12.1, confirmando el intercambio completo de mensajes de ping a través del enlace R1–R2.*

***Tramas ICMP encapsuladas en MPLS (filtro `mpls`)***




*Al aplicar el filtro `mpls`, solo se muestran los **ICMP Echo Request** enviados por R1 hacia R3. En el detalle de los paquetes se observa la cabecera **"MultiProtocol Label Switching Header, Label: 17"**, lo que indica que estos Echo Request salen encapsulados con una etiqueta MPLS cuando atraviesan el enlace FastEthernet0/0 entre R1 y R2.*

***¿Qué paquete sale encapsulado en MPLS? ¿el ICMP Echo Request o el ICMP Echo Reply? ¿ Por qué?***

El paquete que sale encapsulado en MPLS es el ICMP Echo Request, es decir, el ping que va desde R1 (172.16.12.1) hacia la loopback de R3 (172.16.3.1).

En la captura con filtro icmp se observan tanto los Echo Request como los Echo Reply, pero al aplicar el filtro mpls solo aparecen los Echo Request, y en el detalle del paquete se ve la cabecera "MultiProtocol Label Switching Header" con la etiqueta 17.

El ICMP Echo Reply no se ve encapsulado en MPLS en este segmento porque el router R2 actúa como penultimate hop y aplica PHP (Penultimate Hop Popping), eliminando la etiqueta MPLS antes de reenviar la respuesta hacia R1. Por eso, únicamente los paquetes de ida (Echo Request) se observan con etiqueta MPLS, mientras que los paquetes de vuelta (Echo Reply) llegan a R1 como tráfico IP normal.

## 4 Conclusión

A lo largo de la práctica se diseñó, configuró y analizó una red MPLS completa, partiendo del plano IP y llegando al estudio del comportamiento de las etiquetas a nivel de trama. Primero se armó la topología y se configuró el direccionamiento IP y OSPF, verificando la conectividad extremo a extremo mediante pings, traceroute y la tabla de rutas. Luego se comprobó el funcionamiento de CEF, que provee la tabla de reenvío sobre la cual MPLS aplica las etiquetas.

Posteriormente se habilitó MPLS en las interfaces de tránsito y se revisó el estado de las vecindades LDP/TDP con los comandos de diagnóstico (show mpls interfaces, show mpls ldp discovery, show mpls ldp neighbor). A partir de las tablas LIB y LFIB se analizó cómo cada router asigna sus etiquetas locales, qué etiquetas recibe de sus vecinos y cómo se realiza el reenvío por etiqueta, incluyendo el uso de implicit-NULL y el mecanismo de Penultimate Hop Popping.

La migración de TDP a LDP permitió observar el impacto que tiene el protocolo de distribución de etiquetas sobre el establecimiento de vecindades MPLS, y la necesidad de homogeneidad en todo el dominio para que el intercambio de etiquetas funcione correctamente. Finalmente, mediante el análisis de capturas en Wireshark se confirmó qué paquetes ICMP salen encapsulados en MPLS y en qué tramo del recorrido se agrega y se elimina la etiqueta.

En conjunto, la práctica permitió relacionar la teoría de MPLS con resultados concretos de consola y de sniffer: desde la formación de vecindades y tablas de etiquetas hasta la forma en que esas etiquetas afectan el camino real de los paquetes a través de la red.

# 5 Anexos

**Proyecto-MPLS-Parcial02-DiDomenico.gns3project**

Proyecto de GNS3 con la topología utilizada en la práctica (routers R1, R2, R3 y Hub).

Link: https://drive.google.com/file/d/114eTEem7VpCULaTlGjkl3xiTcUDclFJU/view?usp=sharing

**r1-consola.dat**

Backup de la configuración final del router R1 (hostname, direccionamiento IP, OSPF, CEF y MPLS).

Link: https://drive.google.com/file/d/1h8vXnytJoXdSH3IyGmQdH2z3mJqEp-Mk/view?usp=sharing

**r2-consola.dat**

Backup de la configuración final del router R2, incluyendo la migración de TDP a LDP.

Link: https://drive.google.com/file/d/1pgfLbspIASzw4cPPYRO8OCnwoNrhbabE/view?usp=sharing

**r3-consola.dat**

Backup de la configuración final del router R3.

Link: https://drive.google.com/file/d/14ZLqsi7QNgVGM2XzPWyVn7NCnLiD5jUL/view?usp=sharing

**Wireshark.pcapng**

Archivo .pcapng de Wireshark sin filtros aplicados y realizado en el segmento compartido entre R1 y R2 (hub Ethernet). Incluye principalmente tráfico de control de la red: mensajes LDP (Label Distribution Protocol), OSPF Hello, CDP y algunos paquetes TCP. Esta captura se utilizó como base en el Paso 9 para aplicar filtros específicos (icmp y mpls) y analizar el encapsulamiento de los pings en MPLS.

Link: https://drive.google.com/file/d/1-AlJRVSqHVIrKt1jC4PvNBabrly3B4w8/view?usp=sharing