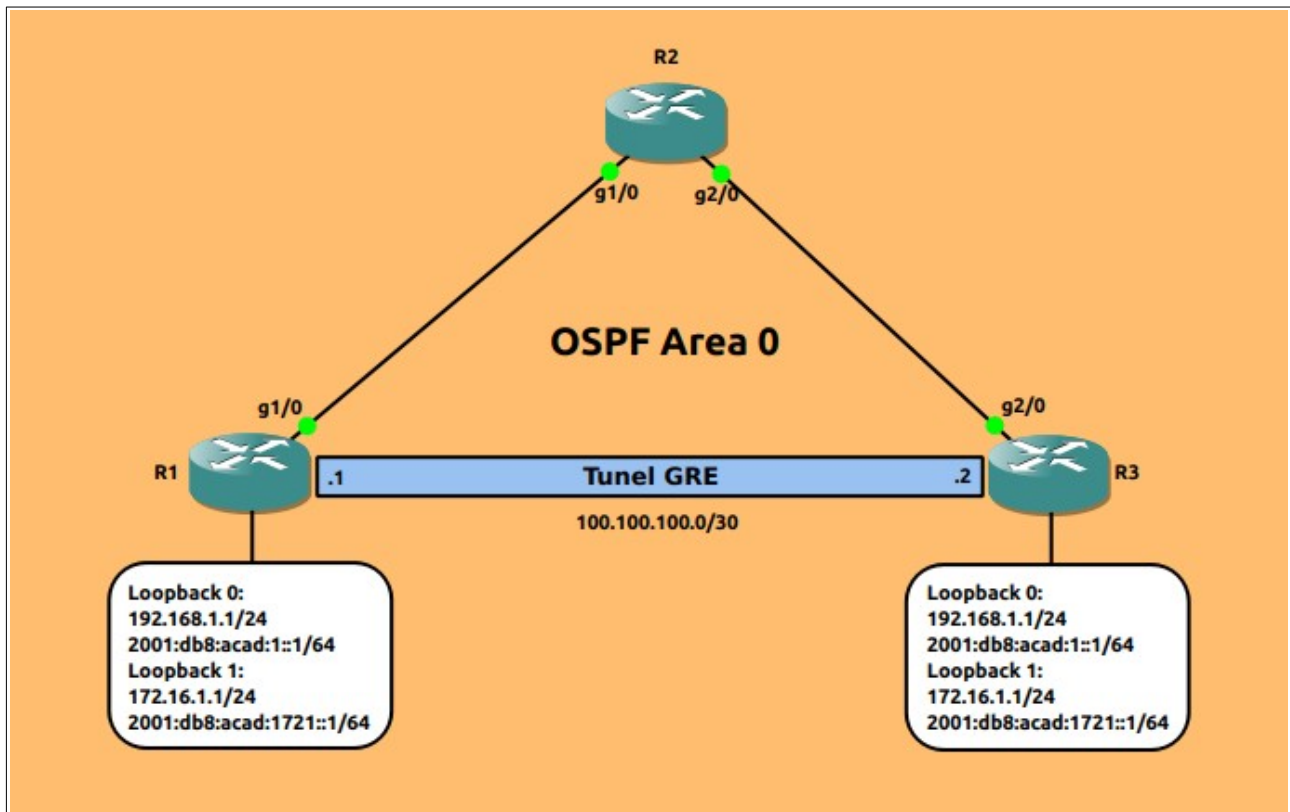


# LABORATORIO

## Túnel GRE

### Topología



### Tabla de direccionamiento

| Dispositivo | Interfaz   | Dirección IPv4 | Dirección IPv6           | IPv6 Link-Local |
|-------------|------------|----------------|--------------------------|-----------------|
| R1          | Gi1/0      | 10.1.2.1/24    | 2001:db8:acad:12::1/64   | fe80::1:1       |
|             | Loopback 0 | 192.168.1.1/24 | 2001:db8:acad:1::1/64    | fe80::1:2       |
|             | Loopback 1 | 172.16.1.1/24  | 2001:db8:acad:1721::1/64 | fe80::1:3       |
| R2          | Gi1/0      | 10.1.2.2/24    | 2001:db8:acad:12::2/64   | fe80::2:1       |
|             | Gi2/0      | 10.2.3.2/24    | 2001:db8:acad:23::2/64   | fe80::2:1       |
| R3          | Gi2/0      | 10.2.3.3/24    | 2001:db8:acad:23::3/64   | fe80::3:1       |
|             | Loopback 0 | 192.168.3.1/24 | 2001:db8:acad:3::1/64    | fe80::3:2       |
|             | Loopback 1 | 172.16.3.1/24  | 2001:db8:acad:1723::1/64 | fe80::3:3       |

## Objetivos del laboratorio

**Parte 1: Armado de la red y configuración básica de los dispositivos.**

**Parte 2: Configuración y verificación de un Tunel GRE con enrutamiento estático.**

**Parte 3: Configuración y verificación de un Tunel GRE con enrutamiento dinámico.**

**Parte 4: Examinar el problema del enrutamiento recursivo con GRE.**

## Introducción

Las **redes overlay** (redes virtuales) permiten introducir flexibilidad en topologías existentes, las cuales pasan a denominarse **redes underlay** (red física subyacente). El protocolo **Generic Routing Encapsulation (GRE – Protocol 47)** de Cisco es una herramienta muy útil que permite crear redes overlay para soportar múltiples propósitos. Es altamente flexible y funciona tanto con **IPv4** como con **IPv6** como red underlay.

En este laboratorio, se implementarán **túneles GRE básicos** sobre redes underlay **IPv4 e IPv6**.

**Nota:** Este laboratorio es un ejercicio de configuración y verificación de diversas implementaciones de túneles **GRE** y no refleja necesariamente las **mejores prácticas** en redes.

## Actividades

**Parte 1: Armado de la red y configuración básica de los dispositivos.**

En esta parte del laboratorio se construye la topología y se realizan las configuraciones básicas de los dispositivos.

**Paso 1: Cableado de los dispositivos.**

Posicionar los dispositivos de acuerdo a como se muestra en la topología y realizar el conexonado de los mismos.

**Paso 2: Configuración de los dispositivos.**

**a)** Conectarse mediante consola a cada uno de los routers, y aplicar la siguiente configuración a cada uno de los dispositivos.

**Router R1**

```
enable
configure terminal
hostname R1
no ip domain lookup
ipv6 unicast-routing
banner motd # R1, Configuración Tunel GRE #
line con 0
```

```
exec-timeout 0 0
logging synchronous
exit
line vty 0 4
  privilege level 15
  password cisco123
  exec-timeout 0 0
  logging synchronous
  login
  exit
router ospf 4
  router-id 1.1.1.4
  exit
ipv6 router ospf 6
  router-id 1.1.1.6
  exit
interface g1/0
  ip address 10.1.2.1 255.255.255.0
  ipv6 address fe80::1:1 link-local
  ipv6 address 2001:db8:acad:12::1/64
  no shutdown
  ip ospf 4 area 0
  ipv6 ospf 6 area 0
  exit
interface loopback 0
  ip address 192.168.1.1 255.255.255.0
  ipv6 address fe80::1:2 link-local
  ipv6 address 2001:db8:acad:1::1/64
  no shutdown
  ip ospf 4 area 0
  ipv6 ospf 6 area 0
  exit
interface loopback 1
  ip address 172.16.1.1 255.255.255.0
  ipv6 address fe80::1:3 link-local
  ipv6 address 2001:db8:acad:1721::1/64
  no shutdown
  exit
end
```

## **Router R2**

```
enable
configure terminal
hostname R2
no ip domain lookup
ipv6 unicast-routing
banner motd # R2, Configuración Tunel GRE #
line con 0
  exec-timeout 0 0
  logging synchronous
  exit
line vty 0 4
  privilege level 15
  password cisco123
  exec-timeout 0 0
  logging synchronous
  login
  exit
router ospf 4
  router-id 2.2.2.4
  exit
ipv6 router ospf 6
  router-id 2.2.2.6
  exit
interface g1/0
  ip address 10.1.2.2 255.255.255.0
  ipv6 address fe80::2:1 link-local
  ipv6 address 2001:db8:acad:12::2/64
  no shutdown
  ip ospf 4 area 0
  ipv6 ospf 6 area 0
  exit
interface g2/0
  ip address 10.2.3.2 255.255.255.0
  ipv6 address fe80::2:2 link-local
  ipv6 address 2001:db8:acad:23::2/64
  no shutdown
  ip ospf 4 area 0
  ipv6 ospf 6 area 0
```

```
exit
end
```

### **Router R3**

```
enable
configure terminal
hostname R3
no ip domain lookup
ipv6 unicast-routing
banner motd # R3, Configuración Tunel GRE #
line con 0
  exec-timeout 0 0
  logging synchronous
exit
line vty 0 4
  privilege level 15
  password cisco123
  exec-timeout 0 0
  logging synchronous
  login
exit
router ospf 4
  router-id 3.3.3.4
exit
ipv6 router ospf 6
  router-id 3.3.3.6
exit
interface g2/0
  ip address 10.2.3.3 255.255.255.0
  ipv6 address fe80::3:1 link-local
  ipv6 address 2001:db8:acad:23::3/64
  no shutdown
  ip ospf 4 area 0
  ipv6 ospf 6 area 0
exit
interface loopback 0
  ip address 192.168.3.1 255.255.255.0
  ipv6 address fe80::3:2 link-local
  ipv6 address 2001:db8:acad:3::1/64
  no shutdown
```

```

ip ospf 4 area 0
ipv6 ospf 6 area 0
exit
interface loopback 1
ip address 172.16.3.1 255.255.255.0
ipv6 address fe80::3:3 link-local
ipv6 address 2001:db8:acad:1723::1/64
no shutdown
exit
end

```

**b) Guardar las configuraciones en startup-config**

```
copy running-config startup-config
```

## Parte 2: Configurar y verificar un Túnel GRE con enrutamiento estático.

En la Parte 2, se configurarán y verificarán túneles GRE entre R1 y R3, utilizando rutas estáticas para la accesibilidad overlay y enrutamiento dinámico para la accesibilidad underlay. Se configurarán dos túneles: uno para tráfico IPv4 y otro para tráfico IPv6. Los túneles GRE son extremadamente flexibles y ofrecen numerosas opciones de implementación más allá de las que se abordan en este laboratorio.

### Paso 1: Verificar conectividad entre R1 y R3.

**a)** Desde el router R1, hacer ping a la interfaz IPv4 Loopback0 del router R3. Todos los pings deberían ser exitosos.

```

R1# ping 192.168.3.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/28/40 ms

```

**b)** Ahora, desde el router R1, hacer ping a la interfaz IPv6 Loopback0 del router R3. Todos los pings deberían ser exitosos.

```

R1# ping 2001:db8:acad:3::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:ACAD:3::1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/36/80 ms

```

### Paso 2: Creación de túnel GRE IPv4 entre R1 y R3.

**a)** En R1, crear una interfaz Tunnel 0, asignarle la dirección IP 100.100.100.1/30, como tunnel source la interfaz Loopback 0, y como tunnel destination la dirección IP 192.168.3.1.

```

R1# configure terminal
R1(config)# interface tunnel 0
R1(config-if)# ip address 100.100.100.1 255.255.255.252
R1(config-if)# tunnel source loopback 0
R1(config-if)# tunnel destination 192.168.3.1
R1(config-if)# exit

```

**b) En R1, crear una ruta estática hacia la red 172.16.3.0/24 a través de la interfaz Tunnel 0.**

```

R1(config)# ip route 172.16.3.0 255.255.255.0 tunnel 0
R1(config)# end

```

**c) En R3, crear una interfaz Tunnel 0, asignarle la dirección IP 100.100.100.2/30, como tunnel source la interfaz Loopback 0, y como tunnel destination la dirección IP 192.168.1.1.**

```

R3# configure terminal
R3(config)# interface tunnel 0
R3(config-if)# ip address 100.100.100.2 255.255.255.252
R3(config-if)# tunnel source loopback 0
R3(config-if)# tunnel destination 192.168.1.1
R3(config-if)# exit

```

**d) En R3, crear una ruta estática hacia la red 172.16.1.0/24 a través de la interfaz Tunnel 0.**

```

R3(config)# ip route 172.16.1.0 255.255.255.0 tunnel 0
R3(config)# end

```

**e) En R1, ejecutar el comando show interface tunnel 0 y examinar la salida del mismo.**

```

R1# show interface tunnel 0
Tunnel0 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 100.100.100.1/30
  MTU 17916 bytes, BW 100 Kbit/sec, DLY 50000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 192.168.1.1 (Loopback0), destination 192.168.3.1
  Tunnel Subblocks:
    src-track:
      Tunnel0 source tracking subblock associated with Loopback0
      Set of tunnels with source Loopback0, 1 member (includes
      iterators), on interface <OK>

```

Tunnel protocol/transport GRE/IP

Key disabled, sequencing disabled

Checksumming of packets disabled

Tunnel TTL 255, Fast tunneling enabled

Tunnel transport MTU 1476 bytes

Tunnel transmit bandwidth 8000 (kbps)

Tunnel receive bandwidth 8000 (kbps)

Last input never, output never, output hang never

Last clearing of "show interface" counters 00:29:52

Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0

Queueing strategy: fifo

Output queue: 0/0 (size/max)

5 minute input rate 0 bits/sec, 0 packets/sec

5 minute output rate 0 bits/sec, 0 packets/sec

0 packets input, 0 bytes, 0 no buffer

Received 0 broadcasts (0 IP multicasts)

0 runts, 0 giants, 0 throttles

0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort

0 packets output, 0 bytes, 0 underruns

0 output errors, 0 collisions, 0 interface resets

0 unknown protocol drops

0 output buffer failures, 0 output buffers swapped out

f) Desde R1, realizar ping y traceroute a la dirección IP 172.16.3.1.

R1# **ping 172.16.3.1 source 172.16.1.1**

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 = 24/37/68 ms

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)

1 100.100.100.2 24 msec \* 40 msec

**Paso 2: Creación de túnel GRE IPv6 entre R1 y R3.**

a) En R1, crear una interfaz Tunnel 1, asignarle la dirección IPv6

2001:db8:ffff::1/64, como tunnel source la interfaz Loopback 0, como

tunnel destination la dirección IPv6 2001:db8:acad:3::1, y como tunnel

mode GRE IPv6.



```

R1# configure terminal
R1(config)# interface tunnel 1
R1(config-if)# ipv6 address 2001:db8:ffff::1/64
R1(config-if)# tunnel source loopback 0
R1(config-if)# tunnel destination 2001:db8:acad:3::1
R1(config-if)# tunnel mode gre ipv6
R1(config-if)# exit

```

**b) En R1, crear una ruta estática hacia la red 2001:db8:acad:1723::/64 a través de la interfaz Tunnel 1.**

```

R1(config)# ipv6 route 2001:db8:acad:1723::/64 tunnel 1
R1(config)# end

```

**c) En R3, crear una interfaz Tunnel 1, asignarle la dirección IPv6 2001:db8:ffff::2/64, como tunnel source la interfaz Loopback 0, como tunnel destination la dirección IPv6 2001:db8:acad:1::1, y como tunnel mode GRE IPv6.**

```

R3# configure terminal
R3(config)# interface tunnel 1
R3(config-if)# ipv6 address 2001:db8:ffff::2/64
R3(config-if)# tunnel source loopback 0
R3(config-if)# tunnel destination 2001:db8:acad:1::1
R3(config-if)# tunnel mode gre ipv6
R3(config-if)# exit

```

**d) En R3, crear una ruta estática hacia la red 2001:db8:acad:1721::/64 a través de la interfaz Tunnel 1.**

```

R3(config)# ipv6 route 2001:db8:acad:1721::/64 tunnel 1
R3(config)# end

```

**e) En R1, ejecutar el comando show interface tunnel 1 y examinar la salida del mismo.**

```

R1# show interface tunnel 1
Tunnel1 is up, line protocol is up

Hardware is Tunnel
MTU 1456 bytes, BW 100 Kbit/sec, DLY 50000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation TUNNEL, loopback not set
Keepalive not set

Tunnel source 2001:DB8:ACAD:1::1 (Loopback0), destination
2001:DB8:ACAD:3::1

Tunnel Subblocks:
src-track:

```

Tunnel1 source tracking subblock associated with Loopback0  
Set of tunnels with source Loopback0, 2 members (includes  
iterators), on interface <OK>

**Tunnel protocol/transport GRE/IPv6**

Key disabled, sequencing disabled

Checksumming of packets disabled

Tunnel TTL 255

Path MTU Discovery, age 10 mins, min MTU 1280

**Tunnel transport MTU 1456 bytes**

**Tunnel transmit bandwidth 8000 (kbps)**

**Tunnel receive bandwidth 8000 (kbps)**

Last input 00:02:36, output 00:08:59, output hang never

Last clearing of "show interface" counters 00:10:27

Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0

Queueing strategy: fifo

Output queue: 0/0 (size/max)

5 minute input rate 0 bits/sec, 0 packets/sec

5 minute output rate 0 bits/sec, 0 packets/sec

10 packets input, 1152 bytes, 0 no buffer

Received 0 broadcasts (0 IP multicasts)

0 runs, 0 giants, 0 throttles

0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort

10 packets output, 1152 bytes, 0 underruns

0 output errors, 0 collisions, 0 interface resets

0 unknown protocol drops

0 output buffer failures, 0 output buffers swapped out

**f) Desde R1, realizar ping y traceroute a la dirección IPv6 2001:db8:acad:1723::1.**

R1# **ping 2001:db8:acad:1723::1 source 2001:db8:acad:1721::1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 2001:DB8:ACAD:1723::1, timeout is 2 seconds:

**!!!!!**

**Success rate is 100 percent (5/5), round-trip min/avg/max = 12/20/48 ms**

R1# **traceroute 2001:db8:acad:1723::1**

Type escape sequence to abort.

Tracing the route to 2001:DB8:ACAD:1723::1

**1 2001:DB8:FFFF::2 64 msec 28 msec 40 msec**

### Parte 3: Configurar y verificar un Túnel GRE haciendo uso de enrutamiento dinámico.

En la Parte 3, se configurarán y verificarán túneles GRE entre R1 y R3, utilizando enrutamiento dinámico para la accesibilidad overlay y enrutamiento estático para la accesibilidad underlay. Se configurarán dos túneles: uno para tráfico IPv4 y otro para tráfico IPv6.

#### Paso 1: Remover las interfaces Tunnel 0 y Tunnel 1 en R1 y R3.

En R1 y R3, ejecutar los siguientes comando:

```
configure terminal
no interface tunnel 0
no interface tunnel 1
```

#### Paso 2: Reemplazar la configuración de OSPF en R1, R2 y R3 por enrutamiento estático.

a) En R1, R2 y R3, remover la configuración de OSPF con los siguientes comandos:

```
no router ospf 4
no ipv6 router ospf 6
```

b) En R1 y R3, crear una ruta por defecto, tanto para IPv4 como para IPv6, apuntando a R2.

##### Router R1

```
R1(config)# ip route 0.0.0.0 0.0.0.0 10.1.2.2
R1(config)# ipv6 route ::/0 2001:DB8:ACAD:12::2
```

##### Router R3

```
R3(config)# ip route 0.0.0.0 0.0.0.0 10.2.3.2
R3(config)# ipv6 route ::/0 2001:DB8:ACAD:23::2
```

c) En R2, crear enrutamiento estático, tanto para IPv4 como para IPv6, hacia las redes Loopback 0 de R1 y R3.

```
R2# configure terminal
R2(config)# ip route 192.168.1.0 255.255.255.0 10.1.2.1
R2(config)# ip route 192.168.3.0 255.255.255.0 10.2.3.3
R2(config)# ipv6 route 2001:db8:acad:1::/64 2001:db8:acad:12::1
R2(config)# ipv6 route 2001:db8:acad:3::/64 2001:db8:acad:23::3
```

d) Verificar que R1 tiene alcance con la interfaz Loopback 0 de R3, usando como source la Loopback 0 de R1.

```
R1# ping 192.168.3.1 source loopback 0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.1.1
```

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 16/26/56 ms

```
R1# ping 2001:db8:acad:3::1 source loopback 0
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 2001:DB8:ACAD:3::1, timeout is 2 seconds:

Packet sent with a source address of 2001:DB8:ACAD:1::1

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 16/24/36 ms

### Paso 3: Creación de túnel GRE IPv4 entre R1 y R3.

a) En R1, crear una interfaz Tunnel 0, asignarle la dirección IP 100.100.100.1/30, bandwidth de 4000 kbps, mtu 1400, como tunnel source la interfaz Loopback 0, y como tunnel destination la dirección IP 192.168.3.1.

```
R1# configure terminal
```

```
R1(config)# interface tunnel 0
```

```
R1(config-if)# ip address 100.100.100.1 255.255.255.252
```

```
R1(config-if)# bandwidth 4000
```

```
R1(config-if)# ip mtu 1400
```

```
R1(config-if)# tunnel source loopback 0
```

```
R1(config-if)# tunnel destination 192.168.3.1
```

```
R1(config-if)# exit
```

b) En R1, configurar OSPFv2, process-id 4, y router-id 1.1.1.4, habilitar la interfaz Tunnel 0 en Area 0 y la interfaz Loopback 1 en Area 1 (utilizar ya sea el comando network o habilitar OSPF directamente en la interfaz).

```
R1# configure terminal
```

```
R1(config)# router ospf 4
```

```
R1(config-router)# router-id 1.1.1.4
```

```
R1(config-router)# network 100.100.100.0 0.0.0.3 area 0
```

```
R1(config-router)# network 172.16.1.0 0.0.0.255 area 1
```

```
R1(config)# end
```

c) En R3, crear una interfaz Tunnel 0, asignarle la dirección IP 100.100.100.2/30, bandwidth de 4000 kbps, mtu 1400, como tunnel source la interfaz Loopback 0, y como tunnel destination la dirección IP 192.168.1.1.

```
R3# configure terminal
```

```
R3(config)# interface tunnel 0
```

```
R3(config-if)# ip address 100.100.100.2 255.255.255.252
```

```
R3(config-if)# bandwidth 4000
```

```
R3(config-if)# ip mtu 1400
```

```
R3(config-if)# tunnel source loopback 0
```

```
R3(config-if)# tunnel destination 192.168.1.1
```

```
R3(config-if)# exit
```

d) En R3, configurar OSPFv2, process-id 4, y router-id 3.3.3.4, habilitar la interfaz Tunnel 0 en Area 0 y la interfaz Loopback 1 en Area 1 (utilizar ya sea el comando network o habilitar OSPF directamente en la interfaz).

```
R3(config)# router ospf 4
```

```
R3(config-router)# router-id 3.3.3.4
```

```
R3(config-router)# network 100.100.100.0 0.0.0.3 area 0
```

```
R3(config-router)# network 172.16.3.0 0.0.0.255 area 1
```

```
R3(config-router)# end
```

e) En R1, ejecutar el comando show interface tunnel 0 y examinar la salida del mismo.

```
R1# show interface tunnel 0
```

```
Tunnel0 is up, line protocol is up
```

```
Hardware is Tunnel
```

```
Internet address is 100.100.100.1/30
```

```
MTU 17916 bytes, BW 4000 Kbit/sec, DLY 50000 usec,
```

```
reliability 255/255, txload 1/255, rxload 1/255
```

```
Encapsulation TUNNEL, loopback not set
```

```
Keepalive not set
```

```
Tunnel source 192.168.1.1 (Loopback0), destination 192.168.3.1
```

```
Tunnel Subblocks:
```

```
src-track:
```

```
Tunnel0 source tracking subblock associated with Loopback0
```

```
Set of tunnels with source Loopback0, 1 member (includes  
iterators), on interface <OK>
```

```
Tunnel protocol/transport GRE/IP
```

```
Key disabled, sequencing disabled
```

```
Checksumming of packets disabled
```

```
Tunnel TTL 255, Fast tunneling enabled
```

```
Tunnel transport MTU 1476 bytes
```

```
Tunnel transmit bandwidth 8000 (kbps)
```

```
Tunnel receive bandwidth 8000 (kbps)
```

```
Last input 00:00:07, output 00:00:04, output hang never
```

```
Last clearing of "show interface" counters 00:42:50
```

```
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
```

```
Queueing strategy: fifo
```

```
Output queue: 0/0 (size/max)
```

```
5 minute input rate 0 bits/sec, 0 packets/sec
```

```

5 minute output rate 0 bits/sec, 0 packets/sec
62 packets input, 6452 bytes, 0 no buffer
Received 0 broadcasts (0 IP multicasts)
0 runs, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
199 packets output, 20144 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 unknown protocol drops
0 output buffer failures, 0 output buffers swapped out

```

**f) En R1, ejecutar el comando `show ip route ospf` y verificar que la red 172.16.3.0/24 aparece en la tabla de enrutamiento como una ruta aprendida mediante el protocolo OSPF.**

```

R1# show ip route ospf | begin Gateway
Gateway of last resort is 10.1.2.2 to network 0.0.0.0

172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O IA      172.16.3.1/32 [110/26] via 100.100.100.2, 00:30:36, Tunnel0

```

**g) Desde R1, hacer ping a la dirección IP 172.16.3.1 utilizando como source la dirección IP 172.16.1.1. Todos los pings deberían ser exitosos.**

```

R1# ping 172.16.3.1 source 172.16.1.1
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 = 20/28/40 m

```

#### **Paso 4: Creación de túnel GRE IPv6 entre R1 y R3.**

**a) En R1, crear una interfaz Tunnel 1, asignarle la dirección IPv6 2001:db8:ffff::1/64, bandwidth 4000 kbps, como tunnel source la interfaz Loopback 0, como tunnel destination la dirección IPv6 2001:db8:acad:3::1 y como tunnel mode GRE IPv6.**

```

R1# configure terminal
R1(config)# interface tunnel 1
R1(config-if)# ipv6 address 2001:db8:ffff::1/64
R1(config-if)# bandwidth 4000
R1(config-if)# tunnel source loopback 0
R1(config-if)# tunnel destination 2001:db8:acad:3::1
R1(config-if)# tunnel mode gre ipv6
R1(config-if)# exit

```

**b)** En R1, configurar OSPFv3, process-id 6, router-id 1.1.1.6, habilitar la interfaz Tunnel 0 en Area 0 y la interfaz Loopback 1 en Area 1 configurando directamente en las interfaces.

```
R1(config)# ipv6 router ospf 6  
R1(config-rtr)# router-id 1.1.1.3  
R1(config-rtr)# exit  
R1(config)# interface tunnel 1  
R1(config-if)# ipv6 ospf 6 area 0  
R1(config-if)# exit  
R1(config)# interface loopback 1  
R1(config-if)# ipv6 ospf 6 area 1  
R1(config-if)# end
```

**c)** En R3, crear una interfaz Tunnel 1, asignarle la dirección IPv6 2001:db8:ffff::2/64, bandwidth 4000 kbps, como tunnel source la interfaz Loopback 0, como tunnel destination la dirección IPv6 2001:db8:acad:3::1 y como tunnel mode GRE IPv6.

```
R3# configure terminal  
R3(config)# interface tunnel 1  
R3(config-if)# ipv6 address 2001:db8:ffff::2/64  
R3(config-if)# bandwidth 4000  
R3(config-if)# tunnel source loopback 0  
R3(config-if)# tunnel destination 2001:db8:acad:1::1  
R3(config-if)# tunnel mode gre ipv6  
R3(config-if)# exit
```

**d)** En R3, configurar OSPFv3, process-id 6, router-id 3.3.3.6, habilitar la interfaz Tunnel 0 en Area 0 y la interfaz Loopback 1 en Area 1 configurando directamente en las interfaces.

```
R3(config)# ipv6 router ospf 6  
R3(config-rtr)# router-id 3.3.3.6  
R3(config-rtr)# exit  
R3(config)# interface tunnel 1  
R3(config-if)# ipv6 ospf 6 area 0  
R3(config-if)# exit  
R3(config)# interface loopback 1  
R3(config-if)# ipv6 ospf 6 area 1  
R3(config-if)# end
```

**e)** En R1, ejecutar el comando `show interface tunnel 1` y examinar la salida del mismo.

R1# **show interface tunnel 1**

**Tunnel1 is up, line protocol is up**

Hardware is Tunnel

MTU 1456 bytes, **BW 4000 Kbit/sec**, DLY 50000 usec,  
reliability 255/255, txload 1/255, rxload 1/255

Encapsulation TUNNEL, loopback not set

Keepalive not set

**Tunnel source 2001:DB8:ACAD:1::1 (Loopback0), destination  
2001:DB8:ACAD:3::1**

Tunnel Subblocks:

src-track:

Tunnel1 source tracking subblock associated with Loopback0

Set of tunnels with source Loopback0, 2 members (includes  
iterators), on interface <OK>

**Tunnel protocol/transport GRE/IPv6**

Key disabled, sequencing disabled

Checksumming of packets disabled

Tunnel TTL 255

Path MTU Discovery, ager 10 mins, min MTU 1280

**Tunnel transport MTU 1456 bytes**

**Tunnel transmit bandwidth 8000 (kbps)**

**Tunnel receive bandwidth 8000 (kbps)**

Last input 00:00:00, output 00:00:04, output hang never

Last clearing of "show interface" counters 00:17:19

Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0

Queueing strategy: fifo

Output queue: 0/0 (size/max)

5 minute input rate 0 bits/sec, 0 packets/sec

5 minute output rate 0 bits/sec, 0 packets/sec

41 packets input, 5200 bytes, 0 no buffer

Received 0 broadcasts (0 IP multicasts)

0 runts, 0 giants, 0 throttles

0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort

100 packets output, 12316 bytes, 0 underruns

0 output errors, 0 collisions, 0 interface resets

0 unknown protocol drops

0 output buffer failures, 0 output buffers swapped out

**f) En R1, ejecutar el comando show ipv6 route ospf y verificar que la red 2001:db8:acad:1723::/64 aparece en la tabla de enrutamiento como una ruta aprendida mediante el protocolo OSPF.**



```

R1# show ipv6 route ospf

IPv6 Routing Table - default - 11 entries

Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
        B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
        I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
        EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE -
Destination
        NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
        OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2, l -
LISP

OI 2001:DB8:ACAD:1723::1/128 [110/25]
    via FE80::C803:ECFF:FE52:8, Tunnel1

```

g) Desde R1, hacer ping a la dirección IPv6 2001:db8:acad:1723::1 usando como source la dirección IPv6 2001:db8:acad:1721::1. Todos los pings deberían ser exitosos.

```

R1# ping 2001:db8:acad:1723::1 source 2001:db8:acad:1721::1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 2001:DB8:ACAD:1723::1, timeout is 2
seconds:

!!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 8/16/24 ms

```

#### Parte 4: Examinar el problema del enrutamiento recursivo con GRE.

El **enrutamiento recursivo** (recursive routing) en redes overlay ocurre cuando el router aprende la ruta hacia el destino del túnel (endpoint remoto) a través del propio túnel GRE.

Por ejemplo: si R1 determina, según su tabla de enrutamiento, que la mejor forma de llegar al destino del túnel es **a través del túnel mismo**.

Es necesario tener mucho cuidado al configurar los protocolos de enrutamiento para evitar que esto suceda, ya que cuando ocurre, **la red overlay deja de funcionar** (el túnel se cae).

a) Para demostrar lo fácilmente que esto puede ocurrir, agregar la red 192.168.1.0/24 a la configuración de OSPF en R1.

```

R1(config)# router ospf 4
R1(config-router)# network 192.168.1.0 0.0.0.255 area 0
R1(config-router)# exit

R1#

*Jan 14 14:49:17.345: %OSPF-5-ADJCHG: Process 4, Nbr 3.3.3.4 on Tunnel0
from FULL to DOWN, Neighbor Down: Dead timer expired

R1#

*Jan 14 14:49:45.422: %OSPF-5-ADJCHG: Process 4, Nbr 3.3.3.4 on Tunnel0
from LOADING to FULL, Loading Done

R1#

```

```
*Jan 14 14:50:25.620: %OSPF-5-ADJCHG: Process 4, Nbr 3.3.3.4 on Tunnel0  
from FULL to DOWN, Neighbor Down: Dead timer expired
```

**b) R1 muestra que el dead timer expira y luego la adyacencia intenta reiniciarse.  
Observar qué se está registrando (logging) en R3.**

```
*Jan 14 14:48:53.611: %ADJ-5-PARENT: Midchain parent maintenance for IP  
midchain out of Tunnel0 - looped chain attempting to stack
```

```
R3#
```

```
*Jan 14 14:49:03.507: %TUN-5-RECURDOWN: Tunnel0 temporarily disabled due  
to recursive routing
```

```
*Jan 14 14:49:03.507: %LINEPROTO-5-UPDOWN: Line protocol on Interface  
Tunnel0, changed state to down
```

```
*Jan 14 14:49:03.527: %OSPF-5-ADJCHG: Process 4, Nbr 1.1.1.4 on Tunnel0  
from FULL to DOWN, Neighbor Down: Interface down or detached
```

**c) Como se puede ver, R1 reconoce el problema y hasta indica que existe un problema de enrutamiento recursivo (recursive routing). Para solucionar esto, se debe eliminar la declaración de red (network statement) en R1 y el túnel volverá a levantar.**

```
R1(config)# router ospf 4
```

```
R1(config-router)# no network 192.168.1.0 0.0.0.255 area 0
```

```
R1(config-router)# end
```

```
R1#
```

```
*Jan 14 15:54:29.439: %OSPF-5-ADJCHG: Process 4, Nbr 3.3.3.4 on Tunnel0  
from LOADING to FULL, Loading Done
```