



**Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza**

dar-pr-3

Diseño y administración de redes

Autor 1:	Toral Pallás, Héctor - 798095
Autor 2:	Lahoz Bernad, Fernando - 800989
Autor 3:	Martínez Lahoz, Sergio - 801621
Grado:	Ingeniería Informática
Curso:	2023-2024

9 de octubre de 2023

Índice

1. Pregunta 1	2
1.1. Ping de PC1 a PC4	2
2. Pregunta 2	3
2.1. Ping de PC1 a PC2	5
3. Pregunta 3	6
3.1. Ping de PC1 a PC4	7
4. Pregunta 4	8
4.1. Ping de PC1 a PC4	9
5. Pregunta 5	10
5.1. Ping de PC1 a PC4	11
6. Anexos	12
6.1. Escenario I	12
6.2. Escenario II	13
6.3. Escenario III	15
6.4. Escenario IV	16
6.5. Escenario V	17

01

Pregunta 1

Deberemos entregar el project que hayamos creado con GNS3. Y las capturas que demuestren que funciona correctamente y en el informe poner las instrucciones que hayamos utilizado en OvS, con su correspondiente explicación. Se propone, que una vez que nos aseguremos de que las tablas ARP de PC1 y PC4 están vacías (si es necesario reiniciamos PC1 y PC4), hacer un ping entre ambos PC y capturar en el enlace entre switch comprobando la existencia de tramas STP, ARP e ICMP y comprobando los valores MAC e IP (en caso de haberlo).

Para la configuración de ambos OvS hemos ejecutado los siguientes comandos:

```
1 # Borrar bridges y crear br0
2 ovs-vsctl del-br br0
3 ovs-vsctl del-br br1
4 ovs-vsctl del-br br2
5 ovs-vsctl del-br br3
6 ovs-vsctl add-br br0
7
8 # Anadir puertos a br0
9 ovs-vsctl add-port br0 eth1
10 ovs-vsctl add-port br0 eth2
11 ovs-vsctl add-port br0 eth15
12
13 # Configuración STP
14 ovs-vsctl set br br0 stp_enable=true
15 ovs-vsctl set port eth1 other_config={stp-path-cost=1000}
16 ovs-vsctl set port eth2 other_config={stp-path-cost=1000}
```

Primero, borramos los bridges existentes y creamos el bridge br0 completamente limpio. Después le incorporamos los puertos eth1 (PC1 o PC3), eth2 (PC2 o PC4) y eth15 (el otro OvS). Activamos el protocolo Spanning Tree y ajustamos el coste de ruta STP para todos los puertos, aunque no es del todo necesario.

A cada una de las máquinas VPCs se les ha asignado una dirección IP de la red 192.168.10.0/24 (PCn → 192.168.10.n).

1.1. Ping de PC1 a PC4

Se ha ejecutado un ping de PC1 a PC4 capturando en el interfaz de PC1, PC4 y entre switches. En todas las capturas aparecen mensajes del protocolo STP cada 2 segundos. Los paquetes 16 y 17 de la captura en PC1 corresponden a la pregunta ARP de PC1 solicitando directamente la MAC de PC4 (la ip destino es 192.168.10.4), ya que el escenario es una red local. En el resto de capturas aparece este mismo mensaje ARP justo antes de la comunicación con ICMPs.

El primer request es el paquete 18 de la captura en PC1. Se puede observar que la MAC destino es la que se había recibido en la respuesta ARP (00:50:79:66:68:00). En la captura entre switches corresponde al paquete 16, y en la de PC4 al paquete 15. Si comprobamos los campos de IPv4 de todos ellos, vemos que todos tienen el mismo identificador de conexión, 0x2776, y el TTL intacto, indicando que el mensaje no ha sido encaminado por ningún router.

02

Pregunta 2

Deberemos entregar el project que hayamos creado con GNS3. Y las capturas que demuestren que funciona correctamente y en el informe poner las instrucciones que hayamos utilizado en OvS, con su correspondiente explicación. Se propone, que una vez que nos aseguremos de que las tablas ARP de PC1 y PC2 están vacías (si es necesario reiniciamos PC1 y PC2), hacer un ping desde ambos PC (que pertenecen a diferentes redes IP) y capturar en el enlace entre switch comprobando la existencia de tramas ARP e ICMP y comprobando los valores MAC, IP y TTL de estas últimas.

Para la configuración del OvS-1, se ha reutilizado parte de la configuración anterior, cambiando los siguientes comandos:

```
1 # Borrar bridges y crear br0
2 ...
3 # Anadir puertos a br0
4 ...
5 # Deshabilitar STP y learning switch
6 ovs-vsctl set br br0 stp_enable=false
7 ovs-vsctl set-fail-mode br0 secure
8
9 # Anadir flujos al switch
10 ovs-ofctl add-flow br0 in_port=1,actions=mod_vlan_vid=2,output:3 # A
11 ovs-ofctl add-flow br0 in_port=2,actions=mod_vlan_vid=3,output:3 # B
12 ovs-ofctl add-flow br0 in_port=3,dl_vlan=2,actions=strip_vlan,output:1 # C
13 ovs-ofctl add-flow br0 in_port=3,dl_vlan=3,actions=strip_vlan,output:2 # D
```

En ambos switch se ha deshabilitado el Protocolo STP y se habilitado el modo seguro para introducir los flujos manualmente. En la figura 4 se visualizan los flujos que se han configurado en OvS-1.

Para la configuración del OvS-2 ha sido necesario configurar funciones adicionales de router:

```
1 # Borrar bridges y crear br0
2 ...
3
4 # Asociar IP gateway de cada VLAN y asociar puertos
5 ovs-vsctl add-port br0 gateway1 -- set interface gateway1 type=internal # ofport 1
6 ovs-vsctl add-port br0 gateway2 -- set interface gateway2 type=internal # ofport 2
7
8 ifconfig gateway1 192.168.2.254 netmask 255.255.255.0 up
9 ifconfig gateway2 192.168.3.254 netmask 255.255.255.0 up
10
11 # Se habilita el envio de paquetes
12 sysctl -w net.ipv4.ip_forward=1
13
14 ovs-vsctl add-port br0 eth1 # ofport 3
15 ovs-vsctl add-port br0 eth2 # ofport 4
16 ovs-vsctl add-port br0 eth15 trunk=2,3
17
18 # Deshabilitar STP y learning switch
19 ovs-vsctl set br br0 stp_enable=false
20 ovs-vsctl set-fail-mode br0 secure
21
22 # Anadir flujos al switch
23 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,nw_dst=192.168.2.3,actions=output:3
24 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,actions=mod_vlan_vid=2,output:5
25
26 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,nw_dst=192.168.3.4,actions=output:4
27 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,actions=mod_vlan_vid=3,output:5
28
29 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,nw_dst=192.168.2.254,actions=output:1
30 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,actions=mod_vlan_vid=2,output:5
31
32 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0806,nw_dst=192.168.3.254,actions=output:2
```

```

33 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0806,actions=mod_vlan_vid=3,output:5
34
35 ovs-ofctl add-flow br0 in_port=5,dl_vlan=2,dl_type=0x0806,actions=strip_vlan,output:1,3
36 ovs-ofctl add-flow br0 in_port=5,dl_vlan=2,dl_type=0x0800,nw_dst=192.168.2.3,actions=
    strip_vlan,output:3
37 ovs-ofctl add-flow br0 in_port=5,dl_vlan=2,dl_type=0x0800,nw_dst=192.168.2.254,actions=
    strip_vlan,output:1
38 ovs-ofctl add-flow br0 in_port=5,dl_vlan=2,dl_type=0x0800,nw_dst=192.168.3.254/24,actions=
    strip_vlan,output:1
39
40 ovs-ofctl add-flow br0 in_port=5,dl_vlan=3,dl_type=0x0806,actions=strip_vlan,output:2,4
41 ovs-ofctl add-flow br0 in_port=5,dl_vlan=3,dl_type=0x0800,nw_dst=192.168.3.4,actions=
    strip_vlan,output:4
42 ovs-ofctl add-flow br0 in_port=5,dl_vlan=3,dl_type=0x0800,nw_dst=192.168.3.254,actions=
    strip_vlan,output:2
43 ovs-ofctl add-flow br0 in_port=5,dl_vlan=3,dl_type=0x0800,nw_dst=192.168.2.254/24,actions=
    strip_vlan,output:2
44
45 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.2.1,actions=mod_vlan_vid=2,
    output:5
46 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.2.3,actions=output:3
47
48 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.3.2,actions=mod_vlan_vid=3,
    output:5
49 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.3.4,actions=output:4
50
51 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.2.1,actions=mod_vlan_vid=2,
    output:5
52 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.2.254,actions=output:1
53 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.3.254/24,actions=strip_vlan,
    output:1
54
55 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0800,nw_dst=192.168.3.2,actions=mod_vlan_vid=3,
    output:5
56 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0800,nw_dst=192.168.3.254,actions=output:2
57 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0800,nw_dst=192.168.2.254/24,actions=strip_vlan,
    output:2

```

Para el OvS-2 se han creado dos puertos lógicos, gateway1 (192.168.2.254) y gateway2 (192.168.3.254), que se configuran como interfaces internas para que actúen de routers por defecto para cada VLAN. En la figura 2 se visualizan los diferentes outputs del switch OvS-2.

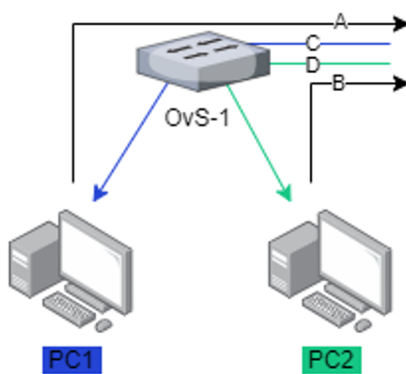


Figura 1: Esquema de flujos en OvS-1

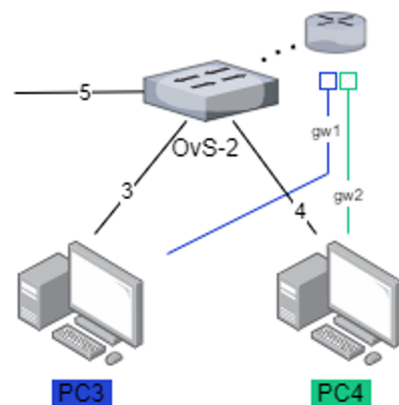


Figura 2: Esquema de outputs en OvS-2

2.1. Ping de PC1 a PC2

Este ping tiene el mismo comportamiento que el de la práctica 3.1:

La comunicación comienza con PC1 preguntando por la MAC del router de su VLAN (192.168.2.254) en el mensaje ARP número 2 de la captura en PC1. Tiene que comunicarse a través de él al ser el destino de diferente VLAN. En la captura entre switches se puede encontrar este mismo ARP en el paquete 3. Se puede observar que se le ha añadido el campo Virtual LAN y que corresponde a un paquete de la VLAN 2. En general, todos los paquetes de esta captura tienen este campo añadido, al tratarse de un segmento entre puertos tagged. El ARP respuesta sigue el camino inverso.

Una vez conocida la MAC (ba:b1:95:f1:93:69), PC1 puede enviar el primer request. Al llegar al router es reenviado por la VLAN 3. En la captura entre switches se puede ver la ida y la vuelta del encaminador (paquetes 5 y 8), cada una con una VLAN diferente. Además, el TTL de la segunda ha disminuido en 1. Se puede comprobar que se trata de la misma conexión viendo el campo identificador de IPv4: 0x235a. Antes de enviar el request de la VLAN 3 es necesario que el router pregunte por la MAC de PC2. El broadcast ARP es el número 6 de la captura entre switches y el 2 de la de PC2.

El paquete reply sigue el camino contrario, volviendo a aparecer 2 veces entre los switches. Esta vez es el paquete de la VLAN 2 el que tiene un TTL menor.

03

Pregunta 3

En este caso no será necesario entregar el project que hayamos creado con GNS3. Sino sólo las capturas que demuestren que funciona correctamente y en el informe poner las instrucciones que hayamos cambiado con respecto al project de la P1, con su correspondiente explicación. Se propone, que una vez que nos aseguremos de que las tablas ARP de PC1 y PC4 están vacías (si es necesario reiniciamos PC1 y PC4), hacer un ping entre ambos PC y capturar en el enlace entre switch comprobando la existencia de tramas GRE donde se encapsulan tramas STP, ARP e ICMP y comprobando los valores MAC e IP (en caso de haberlo).

Para la configuración del OvS-1, se ha reutilizado parte de la configuración de la pregunta 1, cambiando los siguientes comandos:

```
1 # Borrar bridges y crear br0
2 ...
3
4 # Configuración de los puertos y el tunel gre
5 ovs-vsctl add-port br0 eth1
6 ovs-vsctl add-port br0 eth2
7 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.20.2
8 ifconfig eth15 192.168.20.1 netmask 255.255.255.0 broadcast 192.168.20.255 up
9
10 # STP y learning switch activos
11 ...
```

Se configura la interfaz eth15 en OvS-1 para actuar como un túnel punto a punto mediante el protocolo GRE sobre la red IP.

Para la configuración de OvS-2:

```
1 # Borrar bridges y crear br0
2 ...
3
4 ovs-vsctl add-port br0 eth1
5 ovs-vsctl add-port br0 eth2
6 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.20.1
7 ifconfig eth15 192.168.20.2 netmask 255.255.255.0 broadcast 192.168.20.255 up
8
9 # STP y learning switch activos
10 ...
```

El único cambio es la definición de las direcciones IP de cada extremo del túnel.

3.1. Ping de PC1 a PC4

Al igual que en la pregunta 1, la captura entre switches muestra mensajes periódicos del protocolo Spanning Tree, cada 2 segundos. En este caso se puede ver que las tramas ethernet en las que se encapsulan van también encapsuladas en IPv4 a través del túnel GRE. También se observan tramas ARP sin encapsular (paquetes 3, 4, 13 y 14), que sirven para mantener la comunicación IP que sostiene el túnel, y preguntan por las IPs de la red 192.168.20.0/24.

La comunicación producida por el ping comienza con la pregunta ARP del paquete 153 de la captura en PC1, que pregunta directamente por la MAC de PC4, ya que para la máquina es como si estuvieran en la misma LAN. En la captura entre switches se ve puede ver que la trama ARP va encapsulada a través del túnel (paquete 54), y en la captura de PC4 (paquete 165) se observa la trama como si nunca hubiera sido encaminada. La respuesta regresa por el túnel y, una vez recibida, PC1 ya puede enviar el primer ICMP request (paquete 155 de su captura). El recorrido y el contenido del resto de paquetes es análogo al de la pregunta 1, considerando que todas las tramas ICMP están encapsuladas dos veces en IP: una en la comunicación a nivel de red de PC1 y PC4 (red 192.168.10.0/24) y otra en la comunicación a nivel de red del túnel entre switches (red 192.168.20.0/24).

04

Pregunta 4

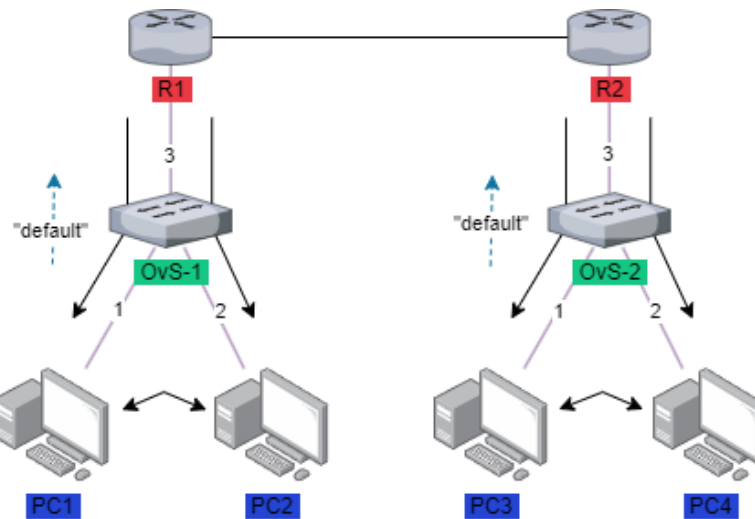
En este caso no será necesario entregar el project que hayamos creado con GNS3. Sino sólo las capturas que demuestren que funciona correctamente y en el informe poner las instrucciones que hayamos cambiado con respecto al project de la P1, con su correspondiente explicación. Se propone, que una vez que nos aseguremos de que las tablas ARP de PC1, PC4, OvS1 y OvS2 están vacías (si es necesario reiniciamos los equipos), hacer un ping entre los PC y capturar en los diferentes enlaces, comprobando la existencia de tramas GRE y cómo se encapsulan tramas STP, ARP e ICMP y comprobando los valores MAC e IP (en caso de haberlo)

Para la configuración del OvS-1, se ha reutilizado parte de la configuración de la pregunta 3, añadiendo los siguientes flujos:

```
1 # Borrar bridges y crear br0
2 ...
3 # Configuración de los puertos y el tunel gre
4 ...
5 # STP activo y learning switch desactivado
6 ovs-vsctl set br br0 stp_enable=true
7 ovs-vsctl set-fail-mode br0 secure
8
9 # Configuración de flujos OvS-1
10 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,actions=output:2,3
11 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,actions=output:1,3
12 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,actions=output:1,2
13
14 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
15 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1
16
17 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1
18 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
19
20 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
21 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
```

```
1 ## Configuración de OvS-gestion-2
2 # Borrar bridges y crear br0
3 ...
4 # Configuración de los puertos y el tunel gre
5 ...
6 # STP activo y learning switch desactivado
7 ...
8
9 # Configuración de flujos OvS-2
10 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,actions=output:2,3
11 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,actions=output:1,3
12 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,actions=output:1,2
13
14 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
15 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1
16
17 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1
18 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
19
20 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
21 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
```

Esta configuración del OvS-2 es la misma que la OvS-2 del ejercicio anterior pero modificando las direcciones



4.1. Ping de PC1 a PC4

En este caso, la información capturada es equivalente a la de la pregunta 3. Como ejemplo de trama ARP para mantener el túnel se puede ver el paquete 3 de la captura entre switches. Como ejemplo de trama ARP encapsulada en el túnel se puede ver el paquete 29 de la captura entre switches, que corresponde al paquete 23 de la captura en PC1, y es el paso previo a la comunicación mediante ping. El primer ICMP request es el paquete 25 de la captura de PC1, que en la captura entre switches es el paquete 31, que está encapsulado dentro del túnel, y que en la captura de PC4, paquete 27, deja de estar encapsulado.

05

Pregunta 5

Deberemos entregar el project que hayamos creado con GNS3. También habrá que entregar las capturas que demuestren que funciona correctamente y en el informe poner las instrucciones que hayamos utilizado tanto en los OvS como en los router, con su correspondiente explicación. Se propone, que una vez que nos aseguremos de que las tablas ARP de PC1 y PC4 están vacías (si es necesario reiniciamos PC1 y PC4), hacer un ping entre ambos PC y capturar en el enlace entre router comprobando la existencia de tramas GRE donde se encapsulan tramas STP, ARP e ICMP y comprobando los valores MAC e IP (en caso de haberlo).

Para la configuración del OvS-1, se ha reutilizado parte de la configuración de la pregunta 4, añadiendo los siguientes comandos:

Listing 1: Configuración de los switches.

```
1 # OvS-1
2 ...
3 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.40.2
4 ifconfig eth15 192.168.20.1 netmask 255.255.255.0 broadcast 192.168.20.255 up
5 route add default gw 192.168.20.254
6
7 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
8 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1
9
10 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1
11 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
12 ...
13
14 # OvS-2
15 ...
16 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.20.1
17 ifconfig eth15 192.168.40.2 netmask 255.255.255.0 broadcast 192.168.40.255 up
18 route add default gw 192.168.40.254
19
20 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
21 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1
22
23 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1
24 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
25 ...
```

Los routers se han configurado de esta manera:

Listing 2: Configuración de R1.

```
1 conf terminal
2 > int FastEthernet 1/0
3 > no sh
4 > ip address 192.168.20.254 255.255.255.0
5 > exit
6 > int FastEthernet 1/1
7 > no sh
8 > ip address 192.168.30.1 255.255.255.0
9 > exit
10 > ip route 192.168.40.2 255.255.255.255 192.168.30.2
11 > exit
12 write
13
```

Listing 3: Configuración de R2.

```
1 conf terminal
2 > int FastEthernet 1/0
3 > no sh
4 > ip address 192.168.40.254 255.255.255.0
5 > exit
6 > int FastEthernet 1/1
7 > no sh
8 > ip address 192.168.30.2 255.255.255.0
9 > exit
10 > ip route 192.168.20.1 255.255.255.255 192.168.30.1
11 > exit
12 write
13
```

5.1. Ping de PC1 a PC4

En esta configuración los paquetes IP sobre los que se monta el túnel entre switches son encaminados por dos routers, por lo que la única diferencia con respecto a los dos anteriores pings es la evolución del TTL de la capa de red más inferior. Si observamos la pregunta ARP previa a la comunicación entre PCs, podemos ver que en las capturas de PC1 (paquete 1) y PC4 (paquete 9), los mensajes viajan por ethernet sin encapsular. Cuando comprobamos ese mismo ARP en la captura entre el OvS-1 y R1 (paquete 19) vemos que viaja encapsulado a través de un túnel GRE que va de la dirección IP 192.168.20.1 (la de OvS-1) a la dirección IP 192.168.40.2 (la de OvS-2), y el paquete IP sobre el que viaja en el túnel tiene un TTL de 64. En la captura entre routers, se puede ver que el nivel IP correspondiente a este paquete (19) tiene un TTL de 63, por haber pasado por R1, y que en la captura entre R2 y OvS-2 (paquete 12) tiene un TTL de 62. El ARP de respuesta sigue el camino inverso y es el paquete capturado entre R1 y OvS-1 (paquete 20) el que tiene 62 de valor TTL.

El primer ICMP request surge en la captura de PC1 (paquete 23) y sigue el mismo camino que el broadcast ARP antes descrito, al igual que lo hacen el resto de ICMP request. Los ICMP reply siguen el camino del ARP de respuesta.

06

Anexos

6.1. Escenario I

Configuración de OvS como Learning Switch

Listing 4: Configuración de OvS-gestion-N.

```
1 #!/bin/sh
2 # Borrar bridges
3 ovs-vsctl del-br br0
4 ovs-vsctl del-br br1
5 ovs-vsctl del-br br2
6 ovs-vsctl del-br br3
7
8 ovs-vsctl add-br br0
9
10 ovs-vsctl add-port br0 eth1
11 ovs-vsctl add-port br0 eth2
12 ovs-vsctl add-port br0 eth15
13
14 # Configuración STP
15 ovs-vsctl set br br0 stp_enable=true
16 ovs-vsctl set port eth1 other_config={stp-path-cost=1000}
17 ovs-vsctl set port eth2 other_config={stp-path-cost=1000}
```

Listing 5: Comandos para la comprobación del funcionamiento.

```
1 ovs-vsctl -- --columns=name,ofport list interface # lista interfaces y puertos
2 ovs-vsctl list interface # lista puertos
3 ovs-vsctl list bridge # lista switches
```

Listing 6: Configuración de las máquinas VPC.

```
1 # PC1
2 ip 192.168.10.1 netmask 255.255.255.0
3 save
4 # PC2
5 ip 192.168.10.2 netmask 255.255.255.0
6 save
7 # PC3
8 ip 192.168.10.3 netmask 255.255.255.0
9 save
10 # PC4
11 ip 192.168.10.4 netmask 255.255.255.0
12 save
```

6.2. Escenario II

Configuración de flujos en OvS para trabajar con VLAN

Listing 7: Configuración de OvS-gestion-1.

```
1 #!/bin/sh
2 # Borrar bridges
3 ovs-vsctl del-br br0
4 ovs-vsctl del-br br1
5 ovs-vsctl del-br br2
6 ovs-vsctl del-br br3
7
8 # Crear bridge limpio y asociar puertos
9 ovs-vsctl add-br br0
10 ovs-vsctl add-port br0 eth1
11 ovs-vsctl add-port br0 eth2
12 ovs-vsctl add-port br0 eth15 trunk=2,3 # Enlace entre switches
13
14 # Deshabilitar STP
15 ovs-vsctl set br br0 stp_enable=false
16 ovs-vsctl set-fail-mode br0 secure
17
18 # Aadir flujos al switch
19 ovs-ofctl add-flow br0 in_port=1,actions=mod_vlan_vid=2,output:3
20 ovs-ofctl add-flow br0 in_port=2,actions=mod_vlan_vid=3,output:3
21 ovs-ofctl add-flow br0 in_port=3,dl_vlan=2,actions=strip_vlan,output:1
22 ovs-ofctl add-flow br0 in_port=3,dl_vlan=3,actions=strip_vlan,output:2
```

Listing 8: Configuración de OvS-gestion-2 (router).

```
1 #!/bin/sh
2 # Borrar bridges
3 ovs-vsctl del-br br0
4 ovs-vsctl del-br br1
5 ovs-vsctl del-br br2
6 ovs-vsctl del-br br3
7
8 # Crear bridge limpio, asociar IP gateway de cada VLAN y asociar puertos
9 ovs-vsctl add-br br0
10 ovs-vsctl add-port br0 gateway1 -- set interface gateway1 type=internal # ofport 1
11 ovs-vsctl add-port br0 gateway2 -- set interface gateway2 type=internal # ofport 2
12
13 ifconfig gateway1 192.168.2.254 netmask 255.255.255.0 up
14 ifconfig gateway2 192.168.3.254 netmask 255.255.255.0 up
15
16 sysctl -w net.ipv4.ip_forward=1
17
18 ovs-vsctl add-port br0 eth1 # ofport 3
19 ovs-vsctl add-port br0 eth2 # ofport 4
20 ovs-vsctl add-port br0 eth15 trunk=2,3
21
22 # Deshabilitar STP
23 ovs-vsctl set br br0 stp_enable=false
24 ovs-vsctl set-fail-mode br0 secure
25
26 #ovs-vsctl -- --columns=name,ofport list interface # Check
27
28 # Aadir flujos al switch
29 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,nw_dst=192.168.2.3,actions=output:3
30 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,actions=mod_vlan_vid=2,output:5
31
32 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,nw_dst=192.168.3.4,actions=output:4
33 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,actions=mod_vlan_vid=3,output:5
34
35 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,nw_dst=192.168.2.254,actions=output:1
36 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,actions=mod_vlan_vid=2,output:5
37
38 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0806,nw_dst=192.168.3.254,actions=output:2
39 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0806,actions=mod_vlan_vid=3,output:5
40
```

```

41 ovs-ofctl add-flow br0 in_port=5,dl_vlan=2,dl_type=0x0806,actions=strip_vlan,output:1,3
42 ovs-ofctl add-flow br0 in_port=5,dl_vlan=2,dl_type=0x0800,nw_dst=192.168.2.3,actions=
    strip_vlan,output:3
43 ovs-ofctl add-flow br0 in_port=5,dl_vlan=2,dl_type=0x0800,nw_dst=192.168.2.254,actions=
    strip_vlan,output:1
44 ovs-ofctl add-flow br0 in_port=5,dl_vlan=2,dl_type=0x0800,nw_dst=192.168.3.254/24,actions=
    strip_vlan,output:1
45
46 ovs-ofctl add-flow br0 in_port=5,dl_vlan=3,dl_type=0x0806,actions=strip_vlan,output:2,4
47 ovs-ofctl add-flow br0 in_port=5,dl_vlan=3,dl_type=0x0800,nw_dst=192.168.3.4,actions=
    strip_vlan,output:4
48 ovs-ofctl add-flow br0 in_port=5,dl_vlan=3,dl_type=0x0800,nw_dst=192.168.3.254,actions=
    strip_vlan,output:2
49 ovs-ofctl add-flow br0 in_port=5,dl_vlan=3,dl_type=0x0800,nw_dst=192.168.2.254/24,actions=
    strip_vlan,output:2
50
51 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.2.1,actions=mod_vlan_vid=2,
    output:5
52 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.2.3,actions=output:3
53
54 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.3.2,actions=mod_vlan_vid=3,
    output:5
55 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.3.4,actions=output:4
56
57 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.2.1,actions=mod_vlan_vid=2,
    output:5
58 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.2.254,actions=output:1
59 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.3.254/24,actions=strip_vlan,
    output:1
60
61 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0800,nw_dst=192.168.3.2,actions=mod_vlan_vid=3,
    output:5
62 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0800,nw_dst=192.168.3.254,actions=output:2
63 ovs-ofctl add-flow br0 in_port=4,dl_type=0x0800,nw_dst=192.168.2.254/24,actions=strip_vlan,
    output:2

```

Listing 9: Configuración de las máquinas VPC.

```

1 # PC1
2 ip 192.168.2.1 netmask 255.255.255.0 192.168.2.254
3 save
4 # PC2
5 ip 192.168.3.2 netmask 255.255.255.0 192.168.3.254
6 save
7 # PC3
8 ip 192.168.2.3 netmask 255.255.255.0 192.168.2.254
9 save
10 # PC4
11 ip 192.168.3.4 netmask 255.255.255.0 192.168.3.254
12 save

```

6.3. Escenario III

Configuración de OvS para conectar LAN mediante WAN con learning switch

Listing 10: Configuración de OvS-gestion-1.

```
1 #!/bin/sh
2 # Borrar bridges
3 ovs-vsctl del-br br0
4 ovs-vsctl del-br br1
5 ovs-vsctl del-br br2
6 ovs-vsctl del-br br3
7
8 ovs-vsctl add-br br0
9 ovs-vsctl add-port br0 eth1
10 ovs-vsctl add-port br0 eth2
11 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.20.2
12 ifconfig eth15 192.168.20.1 netmask 255.255.255.0 broadcast 192.168.20.255 up
13
14 # Configuraci n STP
15 ovs-vsctl set br br0 stp_enable=true
16 ovs-vsctl set port eth1 other_config={stp-path-cost=1000}
17 ovs-vsctl set port eth2 other_config={stp-path-cost=1000}
```

Listing 11: Configuración de OvS-gestion-2.

```
1 #!/bin/sh
2 # Borrar bridges
3 ovs-vsctl del-br br0
4 ovs-vsctl del-br br1
5 ovs-vsctl del-br br2
6 ovs-vsctl del-br br3
7
8 ovs-vsctl add-br br0
9 ovs-vsctl add-port br0 eth1
10 ovs-vsctl add-port br0 eth2
11 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.20.1
12 ifconfig eth15 192.168.20.2 netmask 255.255.255.0 broadcast 192.168.20.255 up
13
14 # Configuraci n STP
15 ovs-vsctl set br br0 stp_enable=true
16 ovs-vsctl set port eth1 other_config={stp-path-cost=1000}
17 ovs-vsctl set port eth2 other_config={stp-path-cost=1000}
```

Listing 12: Configuración de las máquinas VPC.

```
1 #PC1
2 ip 192.168.10.1 netmask 255.255.255.0
3 save
4 #PC2
5 ip 192.168.10.2 netmask 255.255.255.0
6 save
7 #PC3
8 ip 192.168.10.3 netmask 255.255.255.0
9 save
10 #PC4
11 ip 192.168.10.4 netmask 255.255.255.0
12 save
```


6.4. Escenario IV

Configuración de OvS para conectar LAN mediante WAN con definición de flujos.

Listing 13: Configuración de OvS-gestion-1.

```
1 #!/bin/sh
2 # Borrar bridges
3 ovs-vsctl del-br br0
4 ovs-vsctl del-br br1
5 ovs-vsctl del-br br2
6 ovs-vsctl del-br br3
7
8 ovs-vsctl add-br br0
9 ovs-vsctl add-port br0 eth1
10 ovs-vsctl add-port br0 eth2
11 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.20.2
12 ifconfig eth15 192.168.20.1 netmask 255.255.255.0 broadcast 192.168.20.255 up
13
14 ovs-vsctl set br br0 stp_enable=true
15 ovs-vsctl set-fail-mode br0 secure
16
17 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,actions=output:2,3
18 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,actions=output:1,3
19 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,actions=output:1,2
20
21 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
22 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1
23
24 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1
25 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
26
27 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
28 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
```

Listing 14: Configuración de OvS-gestion-2.

```
1 #!/bin/sh
2 # Borrar bridges
3 ovs-vsctl del-br br0
4 ovs-vsctl del-br br1
5 ovs-vsctl del-br br2
6 ovs-vsctl del-br br3
7
8 ovs-vsctl add-br br0
9 ovs-vsctl add-port br0 eth1
10 ovs-vsctl add-port br0 eth2
11 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.20.1
12 ifconfig eth15 192.168.20.2 netmask 255.255.255.0 broadcast 192.168.20.255 up
13
14 ovs-vsctl set br br0 stp_enable=true
15 ovs-vsctl set-fail-mode br0 secure
16
17 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,actions=output:2,3
18 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,actions=output:1,3
19 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,actions=output:1,2
20
21 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
22 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1
23
24 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1
25 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
26
27 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
28 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
```

6.5. Escenario V

Configuración de OvS y SDWAN para conectar LAN mediante WAN con definición de flujos

Listing 15: Configuración de OvS-gestion-1.

```
1 #!/bin/sh
2 # Borrar bridges
3 ovs-vsctl del-br br0
4 ovs-vsctl del-br br1
5 ovs-vsctl del-br br2
6 ovs-vsctl del-br br3
7
8 ovs-vsctl add-br br0
9 ovs-vsctl add-port br0 eth1
10 ovs-vsctl add-port br0 eth2
11 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.40.2
12 ifconfig eth15 192.168.20.1 netmask 255.255.255.0 broadcast 192.168.20.255 up
13 route add default gw 192.168.20.254
14
15 ovs-vsctl set br br0 stp_enable=true
16 ovs-vsctl set-fail-mode br0 secure
17
18 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,actions=output:2,3
19 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,actions=output:1,3
20 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,actions=output:1,2
21
22 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
23 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1
24
25 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1
26 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
27
28 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
29 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
```

Listing 16: Configuración de OvS-gestion-2.

```
1 #!/bin/sh
2 # Borrar bridges
3 ovs-vsctl del-br br0
4 ovs-vsctl del-br br1
5 ovs-vsctl del-br br2
6 ovs-vsctl del-br br3
7
8 ovs-vsctl add-br br0
9 ovs-vsctl add-port br0 eth1
10 ovs-vsctl add-port br0 eth2
11 ovs-vsctl add-port br0 eth15 -- set interface eth15 type=gre options:remote_ip=192.168.20.1
12 ifconfig eth15 192.168.40.2 netmask 255.255.255.0 broadcast 192.168.40.255 up
13 route add default gw 192.168.40.254
14
15 ovs-vsctl set br br0 stp_enable=true
16 ovs-vsctl set-fail-mode br0 secure
17
18 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0806,actions=output:2,3
19 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0806,actions=output:1,3
20 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0806,actions=output:1,2
21
22 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
23 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1
24
25 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1
26 ovs-ofctl add-flow br0 in_port=3,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
27
28 ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
29 ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_dst=192.168.10.0/24,actions=output:3
```

Listing 17: Configuración de R1.

```
1 conf terminal
2 int FastEthernet 1/0
3 no sh
4 ip address 192.168.20.254 255.255.255.0
5 exit
6 int FastEthernet 1/1
7 no sh
8 ip address 192.168.30.1 255.255.255.0
9 exit
10 ip route 192.168.40.2 255.255.255.255 192.168.30.2
11 exit
12 write
```

Listing 18: Configuración de R2.

```
1 conf terminal
2 int FastEthernet 1/0
3 no sh
4 ip address 192.168.40.254 255.255.255.0
5 exit
6 int FastEthernet 1/1
7 no sh
8 ip address 192.168.30.2 255.255.255.0
9 exit
10 ip route 192.168.20.1 255.255.255.255 192.168.30.1
11 exit
12 write
```

Listing 19: Comprobación de funcionamiento para los routers.

```
1 show ip interface brief
2 show ip route
```