

Práctica 2

Redes locales virtuales (VLAN) y puertas de enlace (gateway)

JORGE INFANTE FRAGOSO

Implementación de VLAN

Implementar dos VLAN en un conmutador, en este caso ocupé el conmutador 2960G del segundo rack con dirección IP 172.30.10.12 para conectarse remotamente, decidí ocupar el ID 701 y 702 para las vlan respectivamente.

Las interfaces GigabitEthernet0/1 y 0/2 tienen acceso a la VLAN 701 y las interfaces GigabitEthernet0/3 y 0/4 tienen acceso a la VLAN 702

```
redsw012(config)#vlan 701
redsw012(config-vlan)#name lan701
redsw012(config-vlan)#exit
redsw012(config)#inte
redsw012(config)#interface giga
redsw012(config)#interface gigabitEthernet 0/1
redsw012(config-if)#swi
redsw012(config-if)#switchport mo
redsw012(config-if)#switchport mode ac
redsw012(config-if)#switchport mode access
redsw012(config-if)#swi
redsw012(config-if)#switchport acc
redsw012(config-if)#switchport access vlan 701
redsw012(config-if)#no sh
redsw012(config-if)#exit
redsw012(config)#inte
redsw012(config)#interface gi
redsw012(config)#interface gigabitEthernet 0/2
redsw012(config-if)#sw
redsw012(config-if)#switchport mo
redsw012(config-if)#switchport mode acc
redsw012(config-if)#switchport mode access
redsw012(config-if)#sw
redsw012(config-if)#switchport acc
redsw012(config-if)#switchport access vlan 701
redsw012(config-if)#no sh
redsw012(config-if)#exit
redsw012(config)#
```

```

redsw012(config)#vlan 702
redsw012(config-vlan)#name lan702
redsw012(config-vlan)#exit
redsw012(config)#inte
redsw012(config)#interface giga
redsw012(config)#interface gigabitEthernet 0/3
redsw012(config-if)#swi
redsw012(config-if)#switchport mode
redsw012(config-if)#switchport mode a
redsw012(config-if)#switchport mode access
redsw012(config-if)#swi
redsw012(config-if)#switchport acc
redsw012(config-if)#switchport access vla
redsw012(config-if)#switchport access vlan 702
redsw012(config-if)#no sh
redsw012(config-if)#inte
redsw012(config-if)#interf
redsw012(config-if)#interfac
redsw012(config-if)#exit
redsw012(config)#interface gi
redsw012(config)#interface gigabitEthernet 0/4
redsw012(config-if)#switch
redsw012(config-if)#switchport mode acc
redsw012(config-if)#switchport mode access
redsw012(config-if)#swit
redsw012(config-if)#switchport acce
redsw012(config-if)#switchport access v
redsw012(config-if)#switchport access vlan 702
redsw012(config-if)#no sh
redsw012(config-if)#exit
redsw012(config)#

```

```
redsw012#sh vlan
```

VLAN	Name	Status	Ports
1	default	active	Gi0/5, Gi0/6, Gi0/7, Gi0/8 Gi0/9, Gi0/10, Gi0/11, Gi0/12 Gi0/13, Gi0/14, Gi0/15, Gi0/16 Gi0/17, Gi0/18, Gi0/19, Gi0/20 Gi0/21
216	ssh_switches	active	Gi0/22, Gi0/23, Gi0/24
701	lan701	active	Gi0/1, Gi0/2
702	lan702	active	Gi0/3, Gi0/4
1002	fddi-default	act/unsup	
1003	token-ring-default	act/unsup	
1004	fddinet-default	act/unsup	
1005	trnet-default	act/unsup	

VLAN	Type	SAID	MTU	Parent	RingNo	BridgeNo	Stp	BrdgMode	Trans1	Trans2
1	enet	100001	1500	-	-	-	-	-	0	0
216	enet	100216	1500	-	-	-	-	-	0	0
701	enet	100701	1500	-	-	-	-	-	0	0
702	enet	100702	1500	-	-	-	-	-	0	0

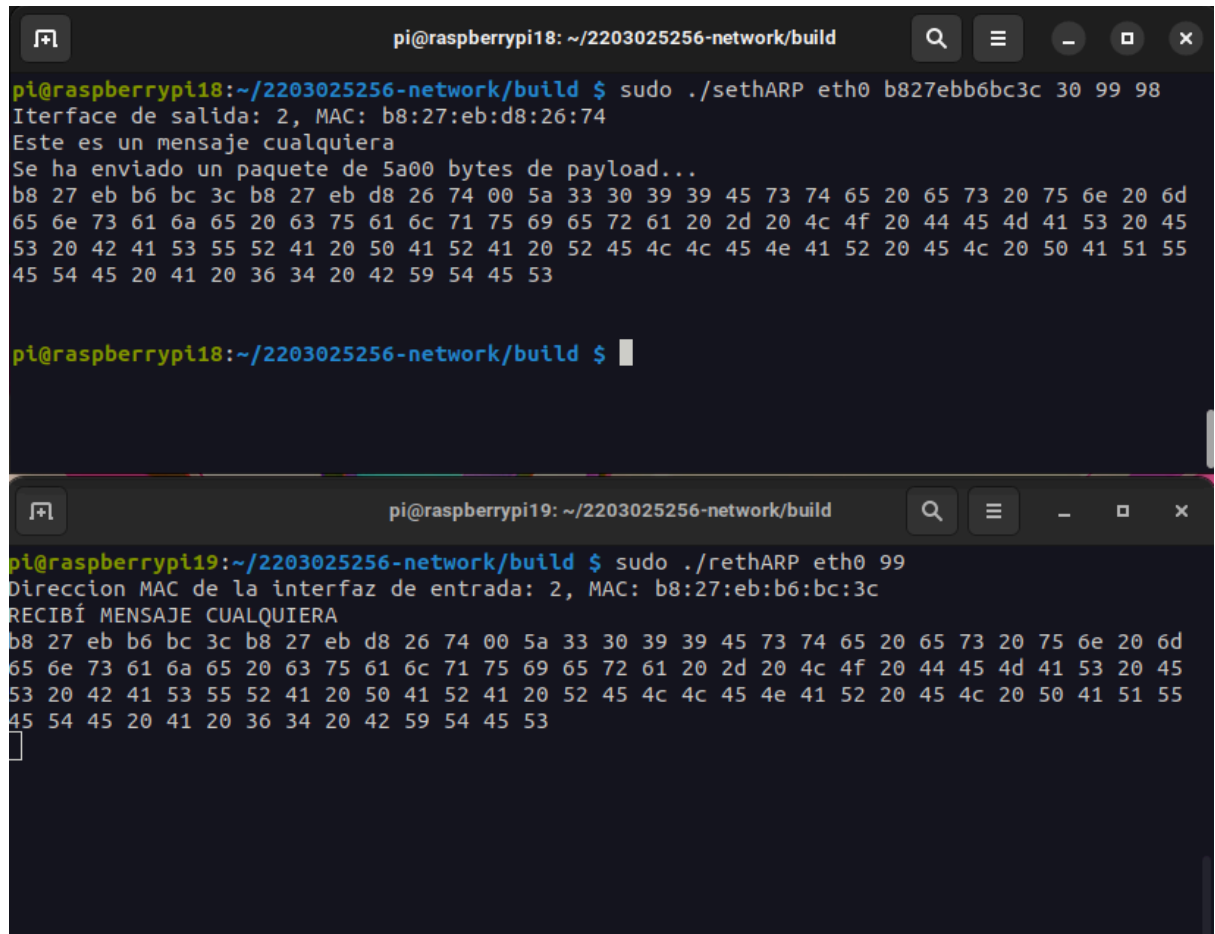
--More--

Preguntas

- En el conmutador conecte sus 4 nodos en cada una de las interfaces asignadas para sus dos VLAN. ¿Qué nodos son los que pueden alcanzarse? ¿Por qué?

Una vez las VLAN son creadas y se les asigna interfaces a cada una, los nodos que comparten una misma VLAN pueden comunicarse entre sí porque comparten un mismo dominio de difusión, pero si se intenta comunicar entre nodos de diferentes VLAN no se va a poder sin ayuda de un enrutador o un dispositivo de capa 3 que permite el enrutamiento.

En este caso los nodos con dirección IP 172.30.2.16 y 172.30.2.17 pertenecen a la VLAN 701 y los nodos con dirección IP 172.30.2.18 172.30.2.19. Por esto, los nodos 16 y 17, así como 18 y 19 se pueden comunicar entre sí, pero 16 y 18 o 17 y 19 no. Por ejemplo: El nodo con dirección IP 172.30.2.18 y 19 sí se pueden comunicar



The image shows two terminal windows. The top window is on Raspberry Pi 18, where the command `sudo ./sethARP eth0 b827ebb6bc3c 30 99 98` is executed. It shows the output of the command, including the interface MAC address and a hex dump of the transmitted packet. The bottom window is on Raspberry Pi 19, where the command `sudo ./rethARP eth0 99` is executed. It shows the output, including the interface MAC address and a hex dump of the received packet. Both hex dumps are identical, indicating successful communication between the two nodes.

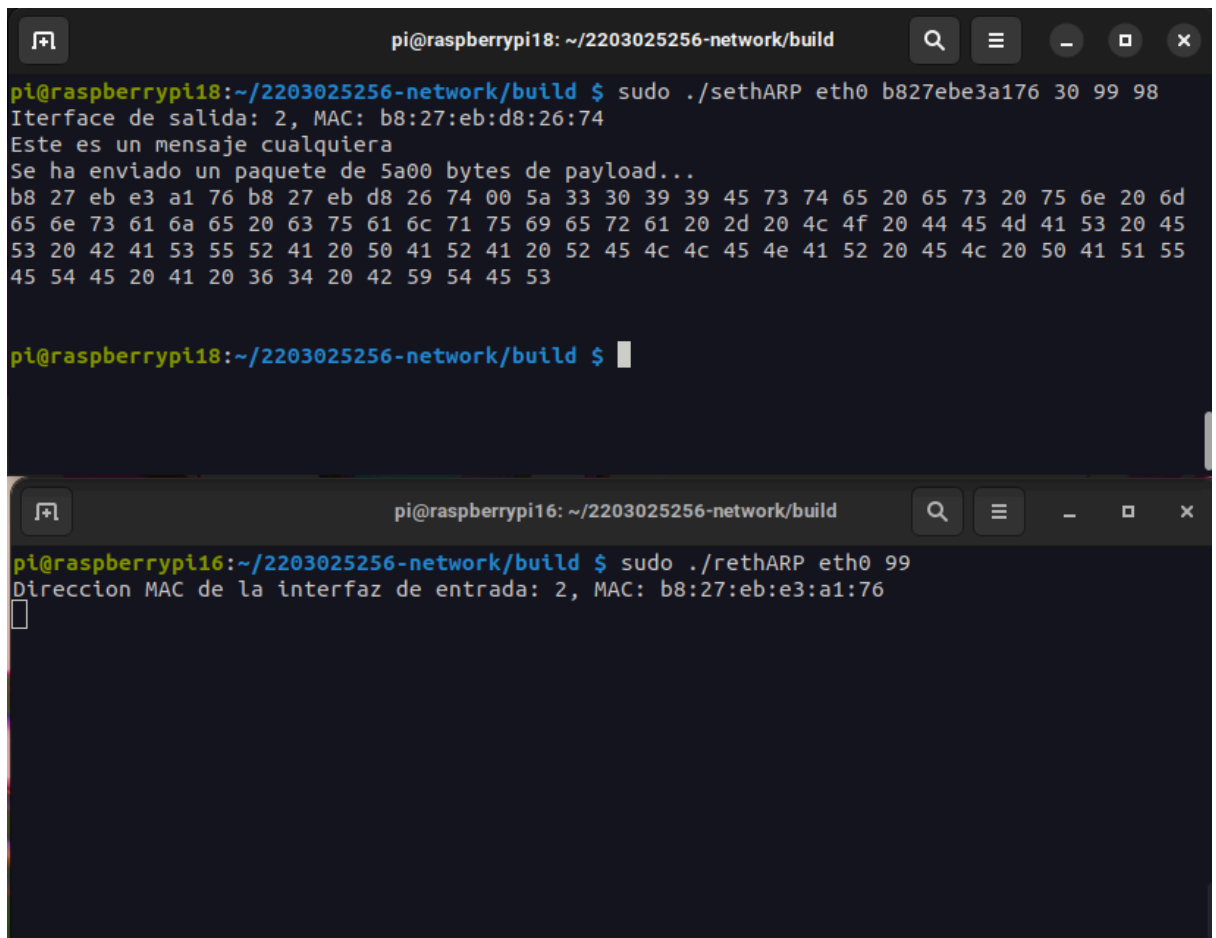
```
pi@raspberrypi18: ~/2203025256-network/build
pi@raspberrypi18:~/2203025256-network/build $ sudo ./sethARP eth0 b827ebb6bc3c 30 99 98
Interface de salida: 2, MAC: b8:27:eb:d8:26:74
Este es un mensaje cualquiera
Se ha enviado un paquete de 5a00 bytes de payload...
b8 27 eb b6 bc 3c b8 27 eb d8 26 74 00 5a 33 30 39 39 45 73 74 65 20 65 73 20 75 6e 20 6d
65 6e 73 61 6a 65 20 63 75 61 6c 71 75 69 65 72 61 20 2d 20 4c 4f 20 44 45 4d 41 53 20 45
53 20 42 41 53 55 52 41 20 50 41 52 41 20 52 45 4c 4c 45 4e 41 52 20 45 4c 20 50 41 51 55
45 54 45 20 41 20 36 34 20 42 59 54 45 53

pi@raspberrypi18:~/2203025256-network/build $

pi@raspberrypi19: ~/2203025256-network/build
pi@raspberrypi19:~/2203025256-network/build $ sudo ./rethARP eth0 99
Direccion MAC de la interfaz de entrada: 2, MAC: b8:27:eb:b6:bc:3c
RECIBÍ MENSAJE CUALQUIERA
b8 27 eb b6 bc 3c b8 27 eb d8 26 74 00 5a 33 30 39 39 45 73 74 65 20 65 73 20 75 6e 20 6d
65 6e 73 61 6a 65 20 63 75 61 6c 71 75 69 65 72 61 20 2d 20 4c 4f 20 44 45 4d 41 53 20 45
53 20 42 41 53 55 52 41 20 50 41 52 41 20 52 45 4c 4c 45 4e 41 52 20 45 4c 20 50 41 51 55
45 54 45 20 41 20 36 34 20 42 59 54 45 53


```

Pero no 16 y 18



The image shows two terminal windows from a Raspberry Pi. The top window, titled 'pi@raspberrypi18: ~/2203025256-network/build', shows the execution of the command 'sudo ./sethARP eth0 b827ebe3a176 30 99 98'. The output indicates that the output interface is 2 with MAC b8:27:eb:d8:26:74, that a message was sent, and that a 5a00 byte payload was transmitted. It then displays a long hex string: b8 27 eb e3 a1 76 b8 27 eb d8 26 74 00 5a 33 30 39 39 45 73 74 65 20 65 73 20 75 6e 20 6d 65 6e 73 61 6a 65 20 63 75 61 6c 71 75 69 65 72 61 20 2d 20 4c 4f 20 44 45 4d 41 53 20 45 53 20 42 41 53 55 52 41 20 50 41 52 41 20 52 45 4c 4c 45 4e 41 52 20 45 4c 20 50 41 51 55 45 54 45 20 41 20 36 34 20 42 59 54 45 53. The bottom window, titled 'pi@raspberrypi16: ~/2203025256-network/build', shows the command 'sudo ./rethARP eth0 99'. The output shows the input interface MAC as b8:27:eb:e3:a1:76, followed by a cursor.

```
pi@raspberrypi18: ~/2203025256-network/build $ sudo ./sethARP eth0 b827ebe3a176 30 99 98
Interface de salida: 2, MAC: b8:27:eb:d8:26:74
Este es un mensaje cualquiera
Se ha enviado un paquete de 5a00 bytes de payload...
b8 27 eb e3 a1 76 b8 27 eb d8 26 74 00 5a 33 30 39 39 45 73 74 65 20 65 73 20 75 6e 20 6d
65 6e 73 61 6a 65 20 63 75 61 6c 71 75 69 65 72 61 20 2d 20 4c 4f 20 44 45 4d 41 53 20 45
53 20 42 41 53 55 52 41 20 50 41 52 41 20 52 45 4c 4c 45 4e 41 52 20 45 4c 20 50 41 51 55
45 54 45 20 41 20 36 34 20 42 59 54 45 53

pi@raspberrypi18: ~/2203025256-network/build $

pi@raspberrypi16: ~/2203025256-network/build $ sudo ./rethARP eth0 99
Direccion MAC de la interfaz de entrada: 2, MAC: b8:27:eb:e3:a1:76
█
```

- b) Si ahora conectamos un par trenzado entre un par de las interfaces de las dos VLAN, ¿qué sucede? ¿Se enlazan los nodos sobrantes? ¿Por qué?
- No, el hecho de conectar un par trenzado entre las dos VLAN no hace que se puedan comunicar ya que cada VLAN son segmentos de red diferentes y requieren de un enrutador o un dispositivo de capa 3 que sea un puente entre las VLAN

Conclusiones

Durante esta práctica comprendí que las VLAN funcionan como una herramienta importante para segmentar la red y aislar el tráfico entre la red, lo que hace que sea beneficioso como para el ancho de banda, para ahorrar recursos y no necesitar más conmutadores, que sea eficiente e incluso que se mejore la seguridad en la red