

# Práctica 3: Enrutamiento estático entre VLAN

JORGE INFANTE FRAGOSO

2203025256

## Resumen

En esta práctica, se crearon y configuraron redes virtuales de área local (VLAN) utilizando un switch multicapa. El objetivo fue segmentar el tráfico entre estas VLAN. El problema es que al conectar nodos a las interfaces asignadas a cada VLAN estos no se pueden comunicar a capa acceso (2) y mucho menos a capa de la red (3). Para habilitar la comunicación entre estas VLAN a nivel de capa 3 se configuró el enrutamiento en el switch multicapa, esto implica la creación de interfaces virtuales para cada VLAN y la asignación de direcciones IP correspondientes. Además, dado que el switch multicapa solo admite enrutamiento estático, se establecieron rutas estáticas para permitir la comunicación entre las redes virtuales.

Al finalizar la práctica se verificó que los nodos dentro de cada VLAN tuvieran comunicación entre sí, así como la capacidad del enrutamiento estático del switch para manejar el tráfico entre las VLAN sin la necesidad de un dispositivo extra de capa 3, lo que permite optimizar recursos.

## Problema

El principal problema de esta práctica es establecer comunicación entre los nodos de redes virtuales de área local (VLAN) diferentes utilizando switches multicapa. El objetivo es configurar adecuadamente las VLAN y el enrutamiento en el switch multicapa para permitir la comunicación entre las redes a nivel de capa de la red.

Las principales actividades que se requieren son: asignar cada VLAN a un segmento de red utilizando direcciones privadas, la asignación de IP a las interfaces virtuales que funcionarán como gateway, permitir el enrutamiento en el switch, la configuración de rutas estáticas para permitir la comunicación entre las VLAN y la configuración de los Host para que funcionen correctamente dentro de su red VLAN.

## Desarrollo y resultados

### Implementación de VLAN

En este primer paso, utilicé el conmutador 2960 del segundo rack que cuenta con la dirección IP 172.30.10.11 para conectarse mediante ssh. Creé 3 VLAN con el identificador 701, 702 y 703 con las direcciones IP 192.168.x.0/24 y las asocié a las interfaces fastEthernet 0/x donde x va desde 1 a 3.

```
redsw011#sh vlan
```

VLAN	Name	Status	Ports
1	default	active	Fa0/4, Fa0/5, Fa0/6, Fa0/7, Fa0/8, Fa0/9, Fa0/10, Fa0/11, Fa0/12, Fa0/13, Fa0/14, Fa0/15, Fa0/16, Fa0/17, Fa0/18, Fa0/19, Fa0/20, Fa0/21, Fa0/22, Fa0/23, Fa0/24, Gi0/1
216	ssh_switches	active	Gi0/2
701	lan701	active	Fa0/1
702	lan702	active	Fa0/2
703	lan703	active	Fa0/3
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
703	enet	100703	1500	-	-	-	-	-	0	0
1002	fddi	101002	1500	-	-	-	-	-	0	0
1003	tr	101003	1500	-	-	-	-	-	0	0
1004	fdnet	101004	1500	-	-	-	ieee	-	0	0
1005	trnet	101005	1500	-	-	-	ibm	-	0	0

## Implementación de las interfaces virtuales del switch

Posteriormente, configuré cada VLAN para que tuviera su interfaz virtual y les asigné la dirección 192.168.x.254 que es la última dirección IP de host de cada bloque de red para que funcionen como gateway.

- Para asignar direcciones IP a las interfaces virtuales de cada VLAN se ocupan los comandos de la figura 3.0:

```
redsw011(config)#interface vlan 701
redsw011(config-if)#ip add
redsw011(config-if)#ip address 192.168.1.254 255.255.255.0
redsw011(config-if)#
```

Figura 3.0

- Una vez se han configurado todas las direcciones IP de las interfaces virtuales, se puede visualizar la configuración con el comando show ip interface brief y se muestra que direcciones IP tiene asignada cada interfaz del conmutador (Figura 3.1).

```
redsw011#sh ip interface brief
```

Interface	IP-Address	OK?	Method	Status	Protocol
Vlan1	unassigned	YES	manual	administratively down	down
Vlan216	172.30.10.11	YES	manual	up	up
Vlan701	192.168.1.254	YES	manual	up	up
Vlan702	192.168.2.254	YES	manual	up	up
Vlan703	192.168.3.254	YES	manual	up	down

Figura 3.1

## Configuración y habilitación del enrutamiento

- Seguidamente, se activa el enrutamiento en el conmutador, primero cambiando la plantilla "default" por la plantilla "lanbase-routing" y después activándolo con el comando "ip routing" (Figura 3.2)

```

redsw011#show sdm prefer
The current template is "lanbase-routing" template.
The selected template optimizes the resources in
the switch to support this level of features for
8 routed interfaces and 255 VLANs.

number of unicast mac addresses:          4K
number of IPv4 IGMP groups + multicast routes: 0.25K
number of IPv4 unicast routes:           4.25K
  number of directly-connected IPv4 hosts: 4K
  number of indirect IPv4 routes:         0.25K
number of IPv4 policy based routing aces: 0
number of IPv4/MAC qos aces:             0.125k
number of IPv4/MAC security aces:        0.375k

redsw011#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
redsw011(config)#ip rout
redsw011(config)#ip routing
redsw011(config)#exit

```

Figura 3.2

## Configuración de los nodos HOST

- Después, se deben configurar los nodos conectados a cada VLAN con las direcciones IP correspondientes a su red para que puedan comunicarse correctamente, la configuración que se utilizó se muestra en la figura 3.3:

```

GNU nano 5.4 /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source /etc/network/interfaces.d/*
auto lo
auto eth0
auto wlan0

iface lo inet loopback

#iface eth0 inet dhcp
iface eth0 inet static
address 192.168.1.1
netmask 255.255.255.0
gateway 192.168.1.254

iface wlan0 inet static
address 172.30.2.11
netmask 255.255.240.0
#gateway 172.30.15.254
wpa-ssid "labred"
wpa-psk e046f323c3280a8a99a852e58dfdee1ac2fe23c0903e2d6fed8f075138fa73e0

```

Figura 3.3

- Dónde address se sustituye por 192.168.x.1, netmask se mantiene igual y el gateway es 192.168.x.254 donde x va desde 1-3 dependiendo la dirección de red en donde se encuentre el nodo
- Posteriormente, se requería que se hicieran rutas estáticas la tabla de enrutamiento a cada VLAN de la siguiente manera:  
`ip route RED/IP_DESTINO MASCARA INTERFAZ|vlan ID_VLAN|DIR_IP`  
Sin embargo, esto no fue necesario ya que en el switch multicapa que se ocupa, si las interfaces VLAN están activadas y los nodos en esas VLAN están configurados

correctamente con las direcciones IP y las puertas de enlace, el switch aprenderá automáticamente las rutas para esas subredes directamente conectadas.

## Resultados

- Finalmente se hace ping desde un nodo a los demás nodos que se encuentran en otras VLAN (Figura 3.4, 3.5)

```
pi@raspberrypi11:~ $ ping 192.168.2.1
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data.
64 bytes from 192.168.2.1: icmp_seq=1 ttl=63 time=1.10 ms
64 bytes from 192.168.2.1: icmp_seq=2 ttl=63 time=1.06 ms
64 bytes from 192.168.2.1: icmp_seq=3 ttl=63 time=1.06 ms
64 bytes from 192.168.2.1: icmp_seq=4 ttl=63 time=1.09 ms
^C
--- 192.168.2.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 1.060/1.080/1.103/0.018 ms
```

Figura 3.4

```
pi@raspberrypi12:~ $ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=63 time=1.14 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=63 time=1.10 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=63 time=1.09 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=63 time=1.10 ms
^C
--- 192.168.1.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 1.092/1.106/1.140/0.019 ms
```

Figura 3.5

## Conclusiones

En esta práctica, logré con éxito la configuración para la comunicación entre VLAN en un switch multicapa habilitando el enrutamiento y configurando las interfaces virtuales del switch. Al momento de habilitar el enrutamiento en el switch multicapa optimizó el uso de recursos al evitar la necesidad de utilizar un enrutador externo, también elimina la necesidad de cablear más, aunque tiene las desventajas que solo permite un número limitado de rutas y son rutas estáticas, dado que no se le pueden aplicar protocolos como OSPF de enrutamiento dinámico, por lo que si se requiere un enrutamiento más complejo o características más avanzadas de seguridad, es mejor optar por un enrutador.

También, es importante saber que los HOSTS raspberry que se ocuparon, pueden fallar y en mi caso tuve que reiniciar las interfaces de red que tienen para que se pudiera ejecutar un ping (Figura 3.6).

```
pi@raspberrypi12:~ $ sudo ifdown eth0
pi@raspberrypi12:~ $ sudo ifup eth0
pi@raspberrypi12:~ $
```

Figura 3.6