

Práctica 4: Configuración de conmutadores y troncales

JORGE INFANTE FRAGOSO

2203025256

Resumen

En esta práctica, se hizo la configuración de conmutadores y troncales para la interconexión de VLANs. El objetivo principal fue crear tres redes VLAN diferentes en dos conmutadores garantizando el aislamiento de cada red. La primera solución es usar cables individuales para cada VLAN, sin embargo no es lo más eficiente en costos por lo que se hizo la configuración de una troncal la cual utiliza un solo cable para el tráfico de todas las VLAN.

La práctica se llevó a cabo utilizando dos conmutadores 2960 y dos 2960G, colocados en 2 racks diferentes, después se conectaron con nodos Raspberry a las primeras interfaces de cada switch. Posteriormente, se crearon tres troncales, una desde el switch 2960 al 2960G del primer rack, la segunda troncal que conecta los dos racks a través de los switches 2960G y por último la troncal que conecta del 2960G al 2960 del segundo rack. La primera troncal permite el tráfico de las 3 VLAN creadas, la segunda troncal conecta las 3 VLAN del primer rack y las 3 del segundo rack, finalmente la tercera troncal permite el tráfico de las 3 VLAN del segundo rack. Finalmente se configuraron interfaces virtuales para cada VLAN que van a actuar como gateway y se habilitó el enrutamiento del switch, esto garantiza que se puedan comunicar nodos que pertenecen a la misma VLAN y que nodos que están en diferente VLAN pero en el mismo rack, las VLAN que se encuentran en el segundo rack no se pueden comunicar con las VLAN del rack uno, se realizaron las pruebas y todo se ejecutó como lo esperado.

Problema

El principal problema de esta práctica es establecer comunicación y garantizar aislamiento entre 3 redes VLAN en cada rack de conmutadores. Este problema se podría solucionar usando un cable por cada VLAN, sin embargo, no es la solución más eficiente en costos y recursos, por lo que es mejor optar por implementar conexiones troncales que permiten la transmisión de datos de las VLAN a través de un único enlace.

Desarrollo y resultados

Implementación de VLAN

En este primer paso, creé 6 VLAN, 3 diferentes en cada rack de conmutadores, con los identificadores 701, 702, 703 para el primer rack (Figura 4.1) y para el segundo 704, 705 y 706 (Figura 4.2)

```
redsw001#sh vlan
```

VLAN	Name	Status	Ports
1	default	active	Fa0/6, Fa0/7, Fa0/8, Fa0/9 Fa0/11, Fa0/13, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gi0/1 Gi0/2
216	ssh_configure	active	
501	lan501	active	
502	lan502	active	
503	lan503	active	
601	lan601	active	Fa0/1
602	lan602	active	
603	lan603	active	
651	lan651	active	
701	lan701	active	Fa0/4
702	lan702	active	
703	lan703	active	

Figura 4.1

```
redsw011#sh vlan
```

VLAN	Name	Status	Ports
1	default	active	Fa0/6, Fa0/7, Fa0/8, Fa0/9 Fa0/11, Fa0/13, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gi0/1 Gi0/2
216	ssh_configure	active	
504	lan504	active	
505	lan505	active	
506	lan506	active	
604	lan604	active	Fa0/1
605	lan605	active	
606	lan606	active	
651	lan651	active	
704	lan704	active	
705	lan705	active	
706	lan706	active	

Figura 4.2

Implementación de las troncales

Para permitir el tráfico de las 3 VLAN sobre un único enlace se implementan las troncales y se les da acceso a la troncal, esto en los 4 switches con sus respectivas VLAN, la figura 4.3 muestra la creación del puerto troncal en la interfaz 12 para el primer switch del segundo rack

```
redsw011(config)#interface fastEthernet 0/12
redsw011(config-if)#sw
redsw011(config-if)#switchport mode tr
redsw011(config-if)#switchport mode trunk
redsw011(config-if)#swi
redsw011(config-if)#switchport tr
redsw011(config-if)#switchport trunk all
redsw011(config-if)#switchport trunk allowed vlan 700
redsw011(config-if)#switchport trunk allowed vlan add 701
redsw011(config-if)#switchport trunk allowed vlan add 702
redsw011(config-if)#
```

Figura 4.3

En el caso de los switches 2960G, se utilizó la interfaz GigabitEthernet 0/13 para configurar la troncal que conecta ambos racks (Figura 4.4) y se colocan todas las VLAN que han sido creadas, incluyendo las que no pertenecen a su rack

```
redsw012(config)#interface gigabitEthernet 0/13
redsw012(config-if)#switchport mode trunk
redsw012(config-if)#switchport trunk allowed vlan 704
redsw012(config-if)#switchport trunk allowed vlan 701
redsw012(config-if)#switchport trunk allowed vlan add 702
redsw012(config-if)#switchport trunk allowed vlan add 703
redsw012(config-if)#switchport trunk allowed vlan add 704
redsw012(config-if)#switchport trunk allowed vlan add 705
redsw012(config-if)#switchport trunk allowed vlan add 706
redsw012(config-if)#exit
redsw012(config)#exit
```

Figura 4.4

Para comprobar que los enlaces troncales han sido creados en las interfaces correspondientes, se hace uso del comando “show interfaces trunk” como se muestra en la Figura 4.5

```
redsw012#sh interfaces trunk

Port      Mode      Encapsulation  Status        Native vlan
Gi0/10    on        802.1q         trunking      1
Gi0/11    on        802.1q         trunking      1
Gi0/12    on        802.1q         trunking      1
Gi0/13    on        802.1q         trunking      1
Gi0/14    on        802.1q         trunking      1
Gi0/15    on        802.1q         trunking      1

Port      Vlans allowed on trunk
Gi0/10    604-606
Gi0/11    601-606
Gi0/12    704-706
Gi0/13    701-706
Gi0/14    504-506
Gi0/15    501-506
```

Figura 4.5

Acceso de las interfaces a las VLAN

En este punto, ya pueden comunicarse nodos que se encuentren en el mismo rack y en la misma VLAN, para esto se asignan direcciones IP a los nodos y se le da acceso a las VLAN a las interfaces conectadas a los nodos. Por ejemplo, si en el rack 1 en el switch 2960 se le da acceso a la interfaz 4 (Figura 4.6) a la VLAN 701 que conecta al nodo 192.168.1.4 y se intenta comunicar con el nodo 192.168.1.5 que se encuentra conectado al switch 2960G y con acceso a la VLAN 701, estos nodos se podrán comunicar eficazmente (Figura 4.6)

```
redsw001(config)#interface fastEthernet 0/4
redsw001(config-if)#sw
redsw001(config-if)#switchport mode acc
redsw001(config-if)#switchport mode access
redsw001(config-if)#swi
redsw001(config-if)#switchport acc
redsw001(config-if)#switchport access vlan 701
redsw001(config-if)#exit
redsw001(config)#exit
```

Figura 4.5

```

pi@raspberrypi04:~ $ ping 192.168.1.5
PING 192.168.1.5 (192.168.1.5) 56(84) bytes of data.
64 bytes from 192.168.1.5: icmp_seq=1 ttl=64 time=1.66 ms
64 bytes from 192.168.1.5: icmp_seq=2 ttl=64 time=1.14 ms
64 bytes from 192.168.1.5: icmp_seq=3 ttl=64 time=1.08 ms
64 bytes from 192.168.1.5: icmp_seq=4 ttl=64 time=1.22 ms
^C
--- 192.168.1.5 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms

```

Figura 4.6

Enrutamiento entre VLAN

Sin embargo, estos nodos solo se pueden comunicar si pertenecen a la misma VLAN, si se desea que se comuniquen los nodos entre VLAN, se deben configurar las interfaces virtuales de cada VLAN para que actúen como gateway y se debe activar el enrutamiento en el switch. En mi caso, escogí las direcciones que van desde 192.168.1.0 con máscara de subred 29 y a cada VLAN se le asignó la última dirección de host de cada bloque de red. (Figura 4.7, 4.8)

```

redsw001(config)#interface vlan 701
redsw001(config-if)#ip add
redsw001(config-if)#ip address 192.168.1.7 255.255.255.248
Bad mask /29 for address 192.168.1.7
redsw001(config-if)#ip address 192.168.1.6 255.255.255.248
redsw001(config-if)#exit
redsw001(config)#inter
redsw001(config)#interface vlan 702
redsw001(config-if)#ip address 192.168.1.14 255.255.255.248
redsw001(config-if)#exit
redsw001(config)#interface vlan 703
redsw001(config-if)#ip address 192.168.1.22 255.255.255.248
redsw001(config-if)#exit
redsw001(config)#exit

```

Figura 4.7

```

redsw011(config)#inter
redsw011(config)#interface vlan 704
redsw011(config-if)#ip add
redsw011(config-if)#ip address 192.168.1.30 255.255.255.248
redsw011(config-if)#exit
redsw011(config)#interface vlan 705
redsw011(config-if)#ip address 192.168.1.38 255.255.255.248
redsw011(config-if)#exit
redsw011(config)#interface vlan 706
redsw011(config-if)#ip address 192.168.1.46 255.255.255.248
redsw011(config-if)#exit
redsw011(config)#exit
redsw011#

```

Figura 4.8

Resultados

Con esto ahora es posible la comunicación entre nodos de diferentes VLAN que se encuentran en el mismo rack de switches, para probarlo se puede ejecutar el comando ping desde los nodos hacia las direcciones de gateway de las VLAN, si no se encuentran en el

mismo rack, no deben funcionar ya que se encuentran aisladas, en este caso, a partir de la dirección 192.168.1.24 ya forma parte del segundo rack (Figura 4.9)

```
pi@raspberrypi04:~ $ ping 192.168.1.14
PING 192.168.1.14 (192.168.1.14) 56(84) bytes of data.
64 bytes from 192.168.1.14: icmp_seq=9 ttl=255 time=8.98 ms

64 bytes from 192.168.1.14: icmp_seq=10 ttl=255 time=2.24 ms
64 bytes from 192.168.1.14: icmp_seq=11 ttl=255 time=2.77 ms
64 bytes from 192.168.1.14: icmp_seq=12 ttl=255 time=1.28 ms
^C
--- 192.168.1.14 ping statistics ---
12 packets transmitted, 4 received, 66.6667% packet loss, time 11302ms
rtt min/avg/max/mdev = 1.277/3.815/8.977/3.027 ms
pi@raspberrypi04:~ $ ping 192.168.1.22
PING 192.168.1.22 (192.168.1.22) 56(84) bytes of data.
64 bytes from 192.168.1.22: icmp_seq=1 ttl=255 time=2.98 ms
64 bytes from 192.168.1.22: icmp_seq=2 ttl=255 time=2.27 ms
64 bytes from 192.168.1.22: icmp_seq=3 ttl=255 time=2.50 ms
^C
--- 192.168.1.22 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 2.268/2.579/2.975/0.294 ms
pi@raspberrypi04:~ $ ping 192.168.1.30
PING 192.168.1.30 (192.168.1.30) 56(84) bytes of data.
From 192.168.1.4 icmp_seq=1 Destination Host Unreachable
From 192.168.1.4 icmp_seq=2 Destination Host Unreachable
From 192.168.1.4 icmp_seq=3 Destination Host Unreachable
^C
```

Figura 4.9

Conclusiones

En esta práctica se realizó con éxito el desafío de establecer comunicación y garantizar aislamiento entre tres redes VLAN en cada rack de conmutadores. La solución a esto fue la creación de troncales, ya que demostró ser eficiente y permite ahorrar recursos en lugar de usar cables individuales. La configuración de las troncales en interfaces específicas permitió el tráfico de todas las VLANs a través de un solo enlace y la asignación de direcciones IP y el acceso de interfaces a las VLANs facilitaron la comunicación entre nodos del mismo rack y de la misma VLAN. Sin embargo, la configuración del enrutamiento entre VLAN permitió la capacidad de comunicación entre nodos de diferentes VLANs y se pudo comprobar la efectividad de la configuración utilizando el comando "ping".