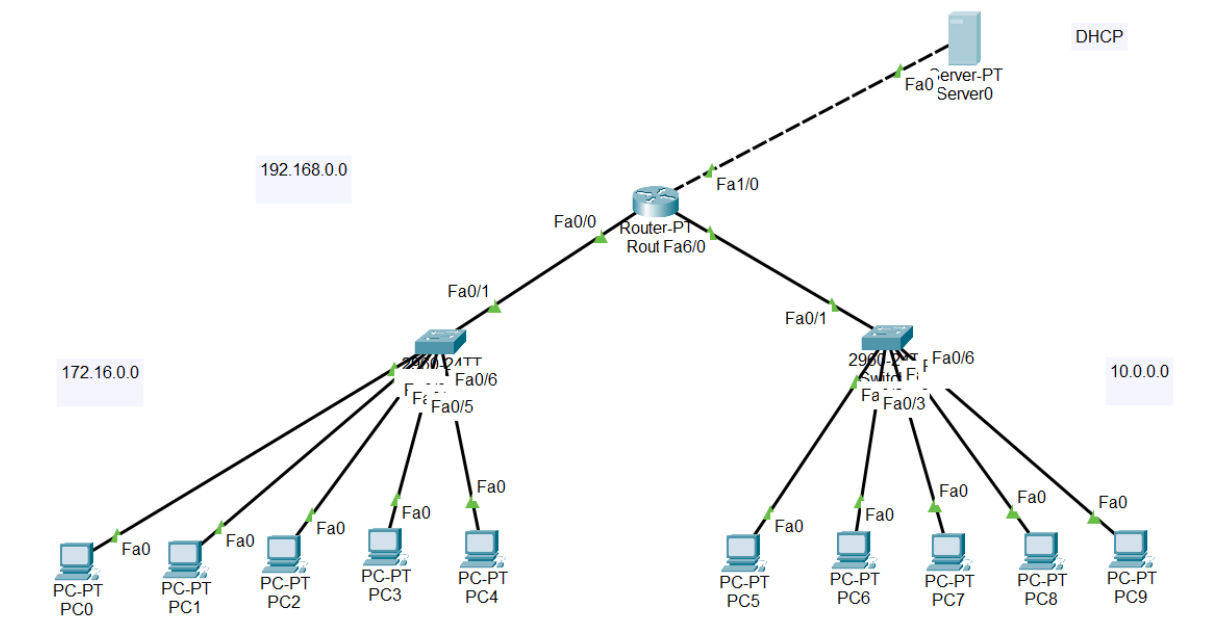


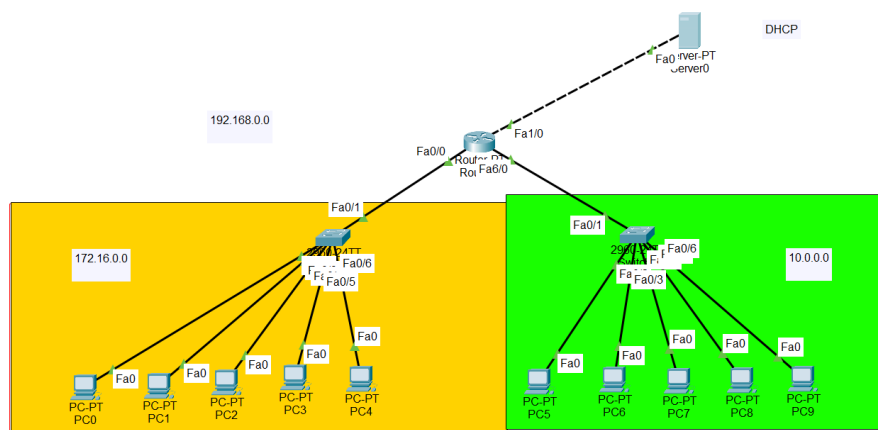
## Explicación de la configuración del router y VLANs

En este laboratorio se realizó la configuración básica de un router Cisco y de las VLANs asociadas para segmentar el tráfico en la red.



Para este lab se hizo esta topología donde se configuraron el DHCP, un firewall por medio de ACL (access list), en el que por medio de dos switches se integraron 10 dispositivos que hicieron un ip hook por medio de el protocolo DHCP, para la VLAN 10 fue 172.16.0.0, para la VLAN 20 10.0.0.0,

Para el gateway se utilizo 192.168.0.0



**Como el segmento principal fue utilizar ACL en el router (101) como firewall se evaluo de la siguiente manera:**

## **1. Configuración inicial del router**

Se accedió al modo privilegiado mediante el comando `enable`, seguido por el modo global de configuración con `conf t`. Una vez en este modo, se configuraron las interfaces físicas del router (por ejemplo, `gig0/0`, `fa0/0`, `fa1/0`, `fa6/0`) asignándoles direcciones IP y máscaras correspondientes a cada red, y activándolas con el comando `no shutdown`. Por ejemplo:

```
int fa0/0
ip address 172.16.0.1 255.255.0.0
no shutdown
exit
```

## **2. Configuración de las estaciones finales (PCs)**

En cada PC conectado a la red, se asignó una dirección IP en el rango correcto de su red, junto con la máscara y la puerta de enlace predeterminada (`default gateway`). Por ejemplo, para un PC en la red `192.168.1.0/24`, se usó `192.168.1.2` como IP y `192.168.1.1` como puerta de enlace.

---

## **3. Configuración de VLANs en los switches**

Se crearon dos VLANs distintas para segmentar el tráfico en el switch:

```
vlan 10
name VLAN10
exit
```

```
vlan 20
name VLAN20
exit
```

Después, se asignaron los puertos correspondientes a cada VLAN utilizando:

```
int fa0/1
```

```

switchport mode access
switchport access vlan 10
exit

```

#### 4. Configuración del DHCP Relay (Helper Address)

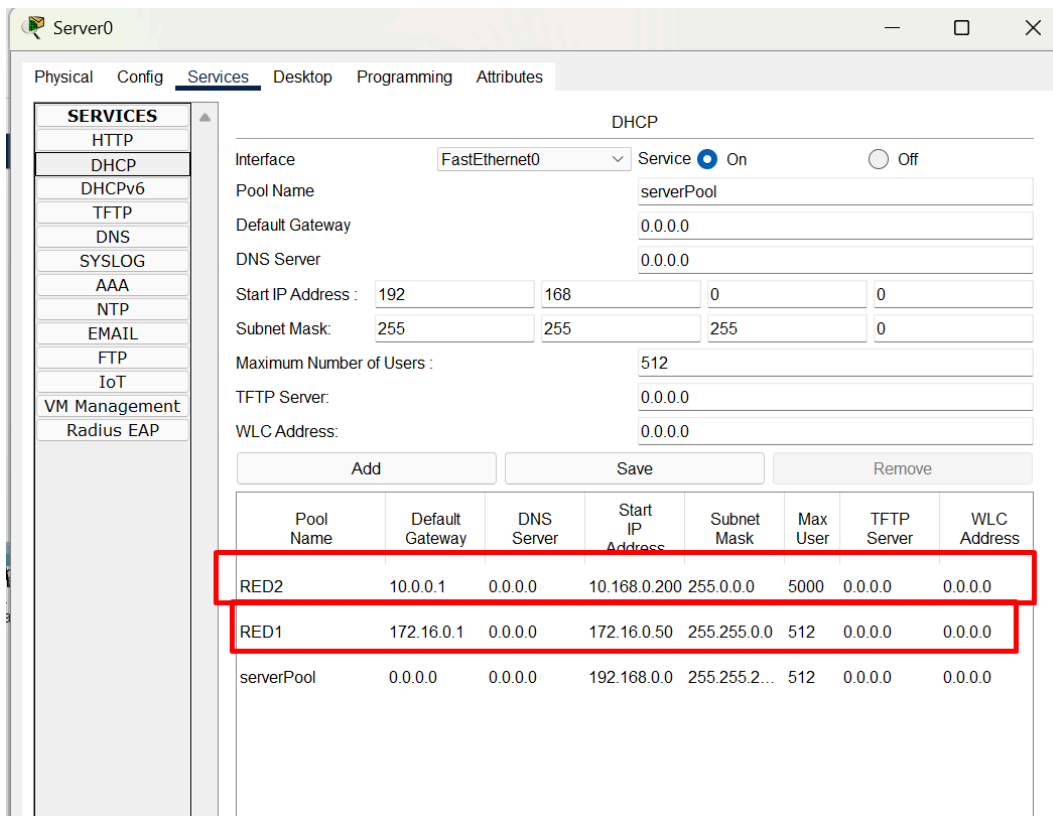
Para permitir que los dispositivos en distintas redes obtengan dirección IP automáticamente, se configuró el **helper-address** en las interfaces del router que sirvieron de puerta de enlace:

```

int fa0/0
ip helper-address 192.168.0.10
exit

```

Se configuró el DHCP también en el servidor para las vlan 10,20 con ip's diferentes:



Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max User	TFTP Server	WLC Address
RED2	10.0.0.1	0.0.0.0	10.168.0.200	255.0.0.0	5000	0.0.0.0	0.0.0.0
RED1	172.16.0.1	0.0.0.0	172.16.0.50	255.255.0.0	512	0.0.0.0	0.0.0.0
serverPool	0.0.0.0	0.0.0.0	192.168.0.0	255.255.2... 0.0	512	0.0.0.0	0.0.0.0

#### 5. Configuración de listas de control de acceso (ACLs)

Qué funciona también como implementación del firewall

Se creó una lista de control de acceso numerada (ACL 101) para filtrar tráfico:

```
access-list 101 deny icmp any any host-unreachable
access-list 101 permit tcp any any eq www
```

Esta lista se aplicó a las interfaces que reciben tráfico:

```
int fa0/0
 ip access-group 101 in
exit
```

---

## 6. Guardar la configuración

Para que la configuración sea permanente, se guardó en la memoria no volátil:

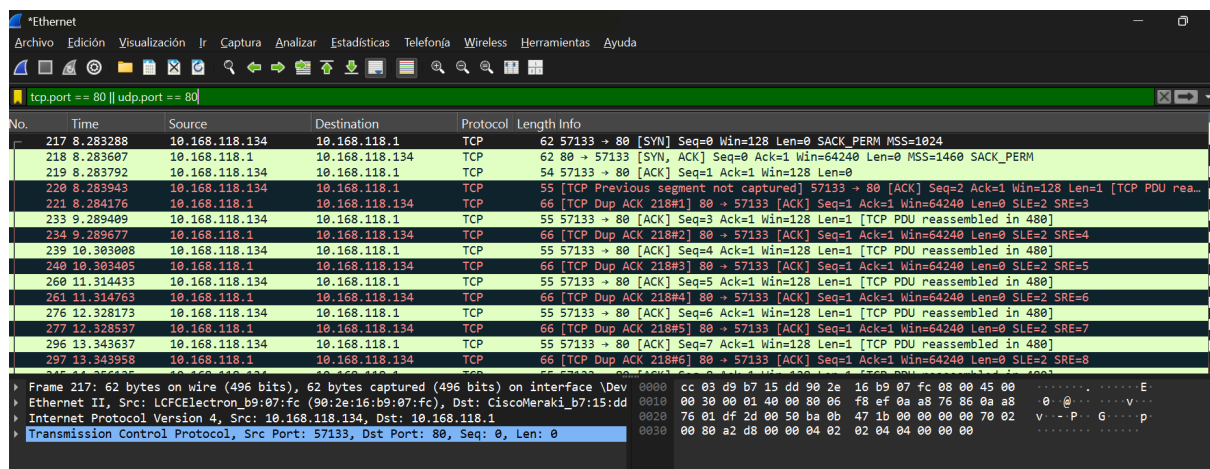
```
copy running-config startup-config
```

---

### Capturas

Para las capturas utilice wireshark, utilice comando ping [www.google.com](http://www.google.com) en el cmd y realice la captura, hubieron servicios de por medio como el dial ya que trabajo con el webex.

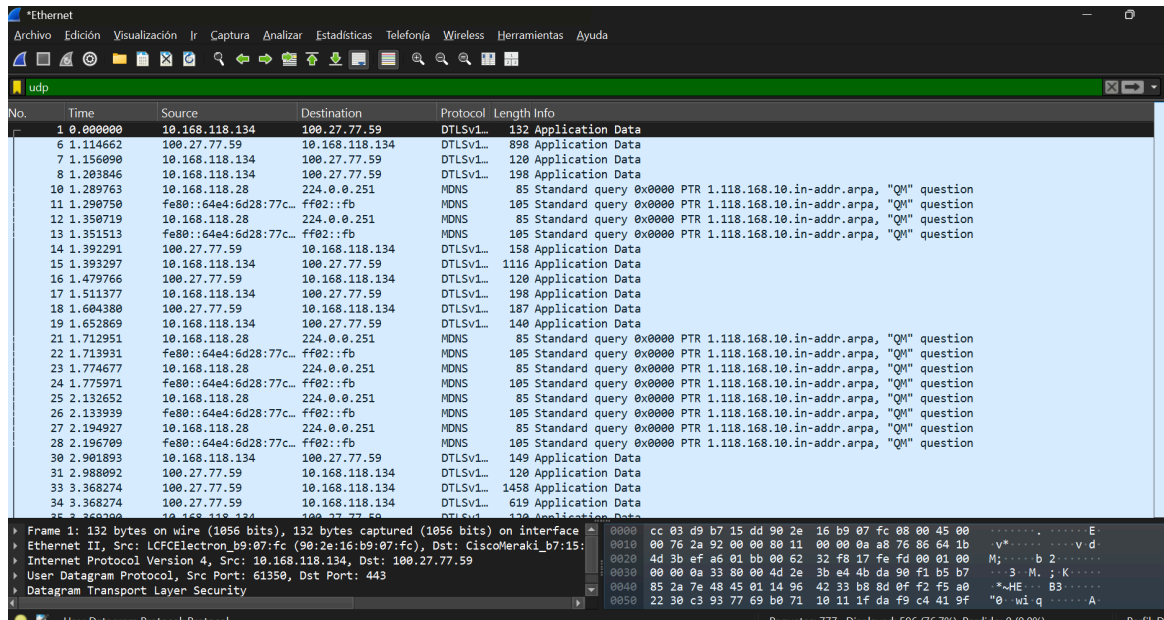
Protocolo TCP donde era port 80 y udp port 80



No.	Time	Source	Destination	Protocol	Length	Info
217	8.283288	10.168.118.134	10.168.118.1	TCP	62	57133 → 80 [SYN] Seq=0 Win=128 Len=0 SACK_PERM MSS=1024
218	8.283607	10.168.118.1	10.168.118.134	TCP	62	80 → 57133 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM
219	8.283792	10.168.118.134	10.168.118.1	TCP	54	57133 → 80 [ACK] Seq=1 Ack=1 Win=128 Len=0
220	8.283943	10.168.118.134	10.168.118.1	TCP	55	[TCP Previous segment not captured] 57133 → 80 [ACK] Seq=2 Ack=1 Win=128 Len=1 [TCP PDU reassembled in 480]
221	8.284176	10.168.118.1	10.168.118.134	TCP	66	[TCP Dup ACK 218#1] 80 → 57133 [ACK] Seq=1 Ack=1 Win=64240 Len=0 SLE=2 SRE=3
223	9.289409	10.168.118.134	10.168.118.1	TCP	55	57133 → 80 [ACK] Seq=3 Ack=1 Win=128 Len=1 [TCP PDU reassembled in 480]
234	9.289677	10.168.118.1	10.168.118.134	TCP	66	[TCP Dup ACK 218#2] 80 → 57133 [ACK] Seq=1 Ack=1 Win=64240 Len=0 SLE=2 SRE=4
239	10.303008	10.168.118.134	10.168.118.1	TCP	55	57133 → 80 [ACK] Seq=4 Ack=1 Win=128 Len=1 [TCP PDU reassembled in 480]
240	10.303405	10.168.118.1	10.168.118.134	TCP	66	[TCP Dup ACK 218#3] 80 → 57133 [ACK] Seq=1 Ack=1 Win=64240 Len=0 SLE=2 SRE=5
260	11.314433	10.168.118.134	10.168.118.1	TCP	55	57133 → 80 [ACK] Seq=5 Ack=1 Win=128 Len=1 [TCP PDU reassembled in 480]
261	11.314763	10.168.118.1	10.168.118.134	TCP	66	[TCP Dup ACK 218#4] 80 → 57133 [ACK] Seq=1 Ack=1 Win=64240 Len=0 SLE=2 SRE=6
276	12.328173	10.168.118.134	10.168.118.1	TCP	55	57133 → 80 [ACK] Seq=6 Ack=1 Win=128 Len=1 [TCP PDU reassembled in 480]
277	12.328537	10.168.118.1	10.168.118.134	TCP	66	[TCP Dup ACK 218#5] 80 → 57133 [ACK] Seq=1 Ack=1 Win=64240 Len=0 SLE=2 SRE=7
296	13.343637	10.168.118.134	10.168.118.1	TCP	55	57133 → 80 [ACK] Seq=7 Ack=1 Win=128 Len=1 [TCP PDU reassembled in 480]
297	13.343958	10.168.118.1	10.168.118.134	TCP	66	[TCP Dup ACK 218#6] 80 → 57133 [ACK] Seq=1 Ack=1 Win=64240 Len=0 SLE=2 SRE=8

Frame 217: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface \Device\NPF{...} Ethernet II, Src: LCFCElectron\_b9:07:fc (90:2e:16:b9:07:fc), Dst: CiscoMeraki\_b7:15:dd (10:20:13:00:b7:15:dd) Internet Protocol Version 4, Src: 10.168.118.134, Dst: 10.168.118.1 Transmission Control Protocol, Src Port: 57133, Dst Port: 80, Seq: 0, Len: 0

## Protocolo UDP evaluado en el filter de wireshark



## Servicio de Chromium para ejecutar el navegador, STREAM UDP evaluado.

