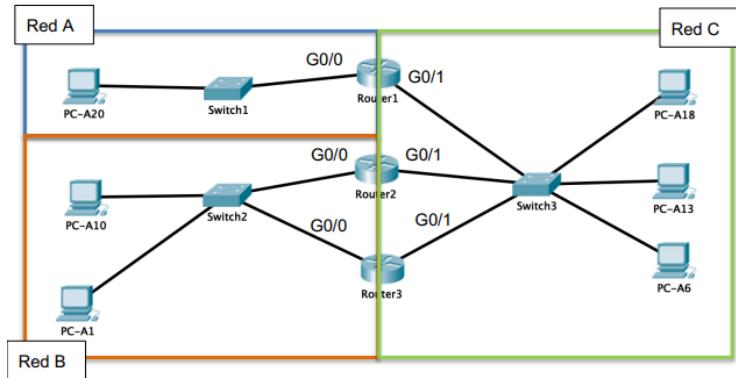


PRÁCTICA 3

1. Planificación Direcciones IPs

Tenemos la red 192.168.25.0/24 y debemos dividirla para cumplir los requisitos específicos de dicha práctica:



- **Subred A (Necesita 3 hosts):**
 - Máscara: /29 (255.255.255.248) -> Nos da 6 hosts útiles.
 - Red: 192.168.25.0
 - Rango: 192.168.25.1 - 192.168.25.6
 - Broadcast 192.168.25.7
- **Subred B (Necesita 5 hosts):**
 - Máscara: /29 (255.255.255.248) -> Nos da 6 hosts útiles.
 - Red: 192.168.25.8 (Siguiente bloque disponible).
 - Rango: 192.168.25.9 - 192.168.25.14
 - Broadcast 192.168.25.15
- **Red C: 172.26.0.0/24 (Ya viene dada).**

2. Configuración del router1:

```
enable
configure terminal
hostname R1          //Asigna el nombre
interface g0/0        // Puerto de entrada del switch1
ip address 192.168.25.2 255.255.255.248 //Asocia dirección de red y Máscara /29
no shutdown      // Habilita el puerto
ipv6 address 2001:db8:acad:1::1/64
no shutdown      // Habilita el puerto
exit            //Salimos de la interfaz
interface g0/1    //Puerto de entrada al switch3
ip address 172.26.0.5 255.255.255.0   //Asocia dirección de red y Máscara /24
no shutdown      // Habilita el puerto
ipv6 address 2001:db8:acad:b::1/64    //Asocia dirección de red y Longitud del prefijo64
no shutdown      // Habilita el puerto
exit //salimos interface
exit //salimos configure terminal
copy running-config startup-config //Guarda la configuración
```

3. Configuración del switch

Switch 1

```
enable
configure
hostname S1          //Asigna el nombre
interface vlan1       //Puerto de entrada de los PCs
ip address 192.168.25.3 255.255.255.248    //IP del Switch 1
no shutdown      //Habilita el puerto
ip default-gateway 192.168.25.2      //Apunta a la IP del router por g0/0
exit
```

```

copy running-config startup-config
Switch 3
enable
configure
hostname S3
interface vlan1
ip address 172.26.0.8 255.255.255.0 //IP del Switch 3
no shutdown
ip default-gateway 172.26.0.5 //IP del R1 por la interfaz g0/1
exit
copy running-config startup-config

```

4. Configuración del PC

PC-A20 (Conectado al Switch 1)

Vas a Desktop -> IP Configuration y metes estos datos exactos:

- IPv4 Address: 192.168.25.1 //IP del PC
- Subnet Mask: 255.255.255.248 //Máscara /29
- Default Gateway: 192.168.25.2 //La IP del Router R1 G0/0
- IPv6 Address: 2001:db8:acad:a::2 //IP del PC ipv6
- Prefix Length: 64 //Longitud del prefijo
- IPv6 Gateway: fe80::1

PC-A18 (Conectado al Router en la otra red)

Vas a Desktop -> IP Configuration:

- IPv4 Address: 172.26.0.3 //IP del PCA-18
- Subnet Mask: 255.255.255.0
- Default Gateway: 172.26.0.5 (La IP del Router R1 G0/1)
- IPv6 Address: 2001:db8:acad:b::2
- Prefix Length: 64
- IPv6 Gateway: fe80::1

5. Comprobación Final

Para documentar que funciona, abre el Command Prompt en el PC-A20 y escribe:

- ping 192.168.25.2 (Debe responder el Router, lado cercano).
- ping 192.168.25.3 (Debe responder el Switch).
- ping 172.26.0.3 (Debe responder el PC-A18, atravesando el router).

6. Comandos útiles de la práctica

Entra en el modo privilegiado (**R1> enable**) y ejecuta estos comandos para ver (y capturar) el estado final:

1. Ver resumen de interfaces (IPv4): **R1# show ip interface brief**
(Esto confirma que G0/0 y G0/1 tienen las IPs correctas y están "up/up").
2. Ver la tabla de enrutamiento IPv4: **R1# show ip route**
(Aquí debes ver las letras "C" (Connectado) y "L" (Local) para tus redes).
3. Ver la tabla de enrutamiento IPv6: **R1# show ipv6 route**
(Para confirmar que el comando ipv6 unicast-routing funcionó).
4. Ver resumen de interfaces (IPv6): **R1# show ipv6 interface brief**
(Verifica que tienes tus direcciones 2001:... y las fe80:...).

5. Ver detalles de la interfaz G0/0: **R1# show ip interface g0/0**
(Muestra detalles técnicos como la MAC y la máscara).
6. Guardar todo (¡Fundamental!): **R1# copy running-config startup-config**
(El comando final que pide el guión para que no se borre al reiniciar).

PRÁCTICA 4

EJERCICIO 1 DHCP

Parte 1: Configurar un router como servidor de DHCP

Paso 1: Configurar las direcciones IPv4 excluidas.

```
R2>enable
```

```
R2#configure terminal
```

```
R2(config)#ip dhcp excluded-address 192.168.10.1 192.168.10.10 -> R2
```

para excluir las primeras 10 direcciones de la LAN R1

```
R2(config)#ip dhcp excluded-address 192.168.30.1 192.168.30.10 -> R2
```

para excluir

las primeras 10 direcciones de la LAN R3

```
R2(config)#exit
```

```
R2#copy running-config startup-config
```

Paso 2: Cree un grupo DHCP en R2 para la LAN R1.

```
R2>enable
```

```
R2#configure terminal
```

```
R2(config)#ip dhcp pool R1-LAN -> Creamos un grupo de DHCP denominado R1-LAN
```

```
R2(dhcp-config)#network 192.168.10.0 255.255.255.0 -> Configuramos la dirección de red
```

```
R2(dhcp-config)#default-router 192.168.10.1 -> Configuramos la puerta de enlace
```

```
R2(dhcp-config)#dns-server 192.168.20.254 -> Configuramos la dirección IP del servidor DNS
```

```
R2(dhcp-config)#exit
```

```
R2#copy running-config startup-config
```

Paso 3: Cree un grupo DHCP en R2 para la LAN R3.

```
R2>enable
```

```
R2#configure terminal
```

```
R2(config)#ip dhcp pool R3-LAN -> Creamos un grupo de DHCP denominado R3-LAN
```

```
R2(dhcp-config)#network 192.168.30.0 255.255.255.0 -> Configuramos la dirección de red
```

```
R2(dhcp-config)#default-router 192.168.30.1 -> Configuramos la puerta de enlace
```

```
R2(dhcp-config)#dns-server 192.168.20.254 -> Configuramos la dirección IP del servidor DNS
```

```
R2(dhcp-config)#exit
```

```
R2#copy running-config startup-config
```

Parte 2: Configurar la retransmisión de DHCP

Paso 1: Configurar R1 y R3 como agente de retransmisión DHCP.

```
R1>enable
```

```
R1#configure terminal
```

```
R1(config)#interface g0/0
```

```
R1(config-if)#ip helper-address 10.1.1.2 -> Configuramos la dirección auxiliar
```

```
R1(config-if)#exit
```

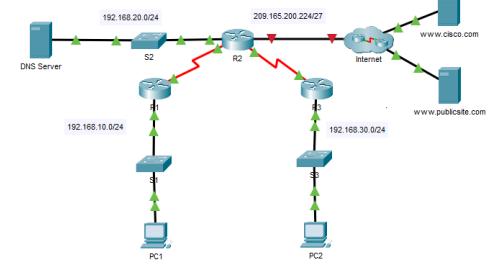
```
R1#copy running-config startup-config
```

```
R3>enable
```

```
R3#configure terminal
```

```
R3(config)#interface g0/0
```

```
R3(config-if)#ip helper-address 10.2.2.2 -> Configuramos la dirección auxiliar
```

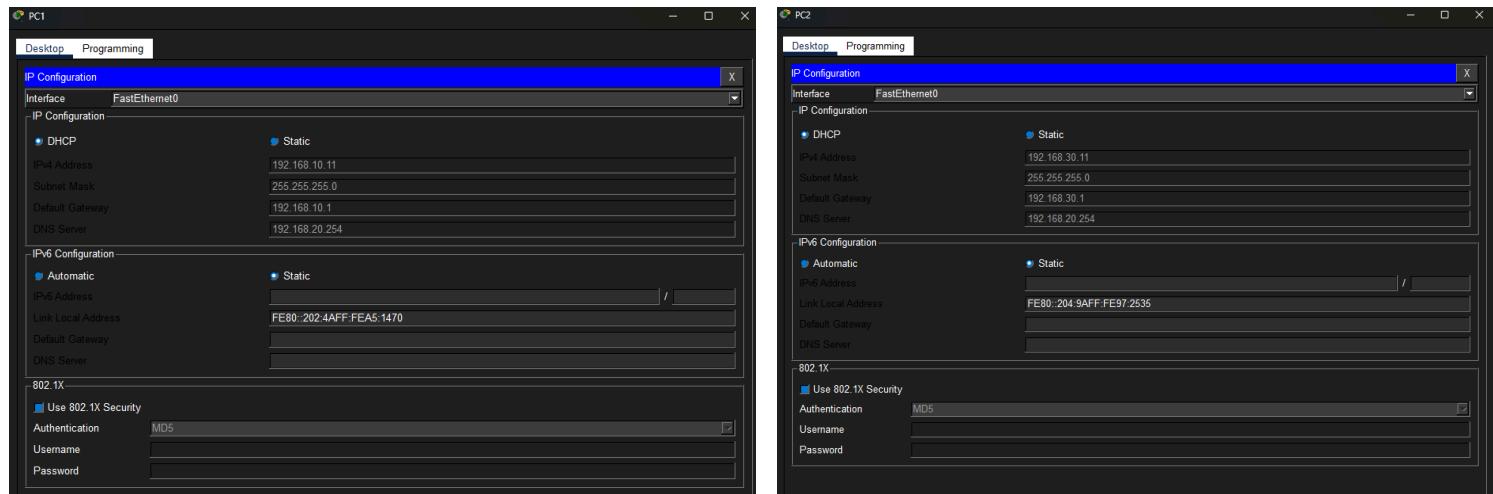


| Dispositivo | Interfaz | Dirección IPv4 | Máscara de subred | Puerta de enlace predeterminada |
|--------------|----------|----------------|-------------------|---------------------------------|
| R1 | G0/0 | 192.168.10.1 | 255.255.255.0 | N/D |
| | S0/0/0 | 10.1.1.1 | 255.255.255.252 | N/D |
| R2 | G0/0 | 192.168.20.1 | 255.255.255.0 | N/D |
| | G0/1 | DHCP asignado | DHCP asignado | N/D |
| | S0/0/0 | 10.1.1.2 | 255.255.255.252 | N/D |
| R3 | S0/0/0 | 10.2.2.2 | 255.255.255.252 | N/D |
| | G0/0 | 192.168.30.1 | 255.255.255.0 | N/D |
| | S0/0/1 | 10.2.2.1 | 255.255.255.0 | N/D |
| PC1 | NIC | DHCP asignado | DHCP asignado | DHCP asignado |
| PC2 | NIC | DHCP asignado | DHCP asignado | DHCP asignado |
| Servidor DNS | NIC | 192.168.20.254 | 255.255.255.0 | 192.168.20.1 |

```
R3(config-if)#exit
R3#copy running-config startup-config
```

Paso 2: Configure los hosts para recibir información de direccionamiento IP de DHCP.

Ir a la configuración de escritorio de PC1 y PC2 y seleccionar DHCP en lugar de Estático para que soliciten la IP.



Parte 3: Configurar un router como cliente DHCP

El router R2 debe obtener su propia dirección IP para la interfaz G0/1 (que simula la conexión a Internet) de un proveedor externo.

R2>enable

R2#configure terminal

R2(config)# interface g0/1

R2(config-if)# ip address dhcp //Este comando habilita a la interfaz para escuchar ofertas DHCP y encenderse
R2(config-if)# no shutdown

Parte 4: Verificar DHCP y la conectividad

Paso 1: Verificar enlaces DHCP Comprobar en el servidor (R2) qué direcciones se han entregado.

R2>enable

R2# show ip dhcp binding

Deberías ver asignaciones en los rangos:

```
R2#show ip dhcp binding
IP address Client-ID/ Lease expiration Type
Hardware address
192.168.10.11 0002.4AA5.1470 --
192.168.30.11 0004.9A97.2535 --
R2#
%DHCP-6-ADDRESS_ASSIGN: Interface GigabitEthernet0/1 assigned DHCP address
209.165.200.231, mask 255.255.255.224, hostname R2
```

```
%DUAL-5-NBRCHANGE: IP-EIGRP 1: Neighbor 209.165.200.225 (GigabitEthernet0/1) is
up: new adjacency
```

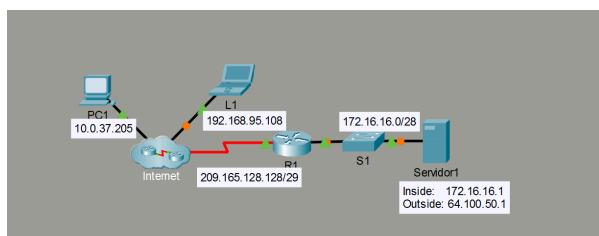
Paso 2: Hacemos ping entre ambos PCs

Comprobamos desde PC1 a PC2 y a la puerta de enlace para asegurar que el enrutamiento y la configuración IP son correctos.

PC1>ping 192.168.30.11

PC2>ping 192.168.10.11

EJERCICIO 2 NAT - ESTÁTICO



Parte 1: Probar el acceso sin NAT, Entender por qué falla (El Diagnóstico)

Antes de configurar, Packet Tracer te pide que pruebes para que veas el error.

- El Ping fallido:** Cuando intentas entrar a 172.16.16.1 desde la PC1 (Internet), falla.

- **¿Por qué?** Porque la dirección 172.16.16.1 es **Privada** (RFC 1918). Los routers de Internet (la nube) no saben cómo llegar a esa IP y descartan el paquete.
2. **El Ping exitoso a R1:** Cuando haces ping a la S0/0/0 de R1, sí funciona.
- **¿Por qué?** Porque esa interfaz tiene una IP Pública. La conexión física existe, solo falta la "traducción" para llegar al servidor de atrás.

Parte 2: configurar NAT estática

Mira tu tabla de direccionamiento o la topología en Packet Tracer.

- IP Privada (Inside Local): 172.16.16.1 (La del Servidor1).
- IP Pública (Inside Global): En esta práctica específica, suele ser 64.100.50.1 (o una dirección similar indicada en las instrucciones/etiquetas del mapa). Asumiremos que es esa para el ejemplo, pero verifícalo en tu mapa.

Paso 1: Mapeo (La regla NAT)

```
R1>enable
R1#configure terminal
R1(config)#ip nat inside source static 172.16.16.1 64.100.50.1 -> asignamos la dirección interna del Servidor1 a su dirección externa (Sintaxis: ip nat inside source static [IP_PRIVADA] [IP_PÚBLICA])
R1(config)#end
R1#copy running-config startup-config
```

(Nota: Asegúrate de que 64.100.50.1 es la IP pública que te pide el ejercicio. Si es otra, cámbiala)

Paso 2: Definir las Puertas (Interfaces)

```
R1>enable
R1#configure terminal
R1(config)#interface g0/0
R1(config-if)#ip nat inside -> Configuramos las interfaces internas (Servidor)
R1(config-if)#exit
R1(config)#interface s0/0/0
R1(config-if)#ip nat outside -> Configuramos las interfaces externas (Internet)
R1(config-if)#end
R1(config)#end
R1#copy running-config startup-config
```

Parte 3: Probar el acceso con NAT

Ahora que el router sabe cómo traducir, volvemos a la PC1 o L1.

1. Ping y Web:

- Ya NO uses la IP 172.16.16.1. Esa es invisible para la PC1.
- Usa la IP Pública: Ping a 64.100.50.1. Debería responder.
- Abre el navegador web en PC1 y pon la URL <http://64.100.50.1>. Debería cargar la página del servidor.

2. Verlo en el Router (Lo que te pide el paso 2):

Vuelve a R1 (en modo privilegiado #) y ejecuta:

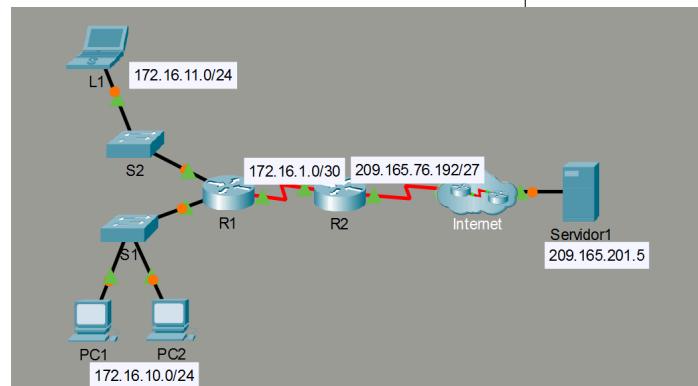
| Pro | Inside global | Inside local | Outside local | Outside global |
|------|----------------|----------------|----------------------|----------------------|
| icmp | 64.100.50.1:10 | 172.16.16.1:10 | 209.165.128.129:10 | 209.165.128.129:10 |
| icmp | 64.100.50.1:11 | 172.16.16.1:11 | 209.165.128.129:11 | 209.165.128.129:11 |
| icmp | 64.100.50.1:12 | 172.16.16.1:12 | 209.165.128.129:12 | 209.165.128.129:12 |
| icmp | 64.100.50.1:13 | 172.16.16.1:13 | 209.165.128.129:13 | 209.165.128.129:13 |
| icmp | 64.100.50.1:14 | 172.16.16.1:14 | 209.165.128.129:14 | 209.165.128.129:14 |
| icmp | 64.100.50.1:15 | 172.16.16.1:15 | 209.165.128.129:15 | 209.165.128.129:15 |
| icmp | 64.100.50.1:16 | 172.16.16.1:16 | 209.165.128.129:16 | 209.165.128.129:16 |
| tcp | 64.100.50.1:80 | 172.16.16.1:80 | 209.165.128.129:1025 | 209.165.128.129:1025 |
| | | | --- | --- |
| | --- | 172.16.16.1 | --- | --- |

```
R1# show ip nat translations
```

- **Inside Global:** La IP pública con la que se presenta el servidor al mundo.
- **Inside Local:** La IP real privada del servidor.
- Esta línea aparecerá siempre, aunque no haya tráfico, porque es **Estática**.

Para ver las estadísticas: R1# show ip nat statistics

```
R1#show ip nat statistics
Total translations: 2 (1 static, 1 dynamic, 1 extended)
Outside Interfaces: Serial0/0/0
Inside Interfaces: GigabitEthernet0/0
Hits: 14 Misses: 9
Expired translations: 8
Dynamic mappings:
```



EJERCICIO 3 NAT - DINÁMICO

Parte 1: configurar NAT dinámica

Paso 1: Configurar el tráfico permitido (La ACL)

Tenemos que decirle al router qué IPs privadas tienen permiso para ser traducidas. La instrucción dice: "cualquier dirección que pertenezca a 172.16.0.0/16".

Calculamos la Wildcard (inversa de la máscara):

- Máscara /16 = 255.255.0.0
- Wildcard = 0.0.255.255

```
R2>enable
```

```
R2#configure terminal
```

```
R2(config)#access-list 1 permit 172.16.0.0 0.0.255.255 -> ACL 1 permite cualquier dirección que pertenezca a 172.16.0.0/16
```

```
R2(config)#exit
```

```
R2#copy running-config startup-config
```

Paso 2: Configurar el conjunto de direcciones (El Pool)

Aquí definimos la "bolsa" de IPs públicas que nos ha dado el proveedor.

- Subred: 209.165.76.196/30
- Rango: .196, .197, .198, .199 (Son 4 direcciones).
- Máscara: /30 es 255.255.255.252.

Vamos a llamar al pool "MIPOOL" (puedes ponerle el nombre que quieras, pero recuerda usar mayúsculas para no liarte).

```
R2(config)# ip nat pool MIPOOL 209.165.76.196 209.165.76.199 netmask 255.255.255.252 (Sintaxis: ip nat pool <NOMBRE> <INICIO> <FIN> netmask <MASCARA>)
```

```
R2>enable
```

```
R2#configure terminal
```

```
R2(config)#ip nat pool NAT_POOL 209.165.76.196 209.165.76.199 netmask  
255.255.255.252 -> para que utilice las cuatro direcciones en el espacio de direcciones  
209.165.76.196/30, se tiene que coger la .196, porque hay un rango de 3 direcciones y no se  
puede pasar de .200  
R2(config)#end  
R2#copy running-config startup-config
```

Respondiendo a la pregunta de la guía: "¿Qué sucedería si más de dos dispositivos intentaran acceder a Internet?"

Como NO estamos usando sobrecarga (PAT), esto es una asignación 1 a 1 temporal. Si tienes 4 IPs en el pool y 5 personas quieren salir a la vez, el quinto usuario se queda sin conexión. Su paquete será descartado porque no quedan IPs públicas libres.

Paso 3: Asociar la ACL con el Pool

Este es el comando que une los pasos 1 y 2.

```
R2(config)# ip nat inside source list 1 pool MIPOOL (Sintaxis: ip nat inside source list <NUM_ACL> pool  
<NOMBRE_POOL>)
```

```
R2>enable  
R2#configure terminal  
R2(config)#ip nat inside source list 1 pool NAT_POOL -> asociar ACL 1 con el conjunto de NAT.  
R2(config)#end  
R2#copy running-config startup-config
```

(Nota importante: Fíjate que NO he puesto la palabra overload al final. Si la pusiera, sería PAT. Sin ella, es NAT Dinámico puro).

Paso 4: Configurar las interfaces (Inside/Outside)

Igual que en el estático, hay que definir las puertas. (Verifica en tu mapa de Packet Tracer cuáles son, pero normalmente en R2 para esta práctica es):

- Serial0/0/0 (o la que va a Internet): outside
- Serial0/0/1 (o la que va a la red interna/otros routers): inside

```
R2>enable  
R2#configure terminal  
R2(config)#interface s0/0/1  
R2(config-if)#ip nat inside -> Configuramos las interfaces con los comandos de NAT  
inside  
R2(config-if)#exit  
R2(config)#interface s0/0/0  
R2(config-if)#ip nat outside -> Configuramos las interfaces con los comandos de NAT  
outside  
R2(config)#end  
R2#copy running-config startup-config
```

Parte 2: verificar la implementación de NAT

Ahora toca generar tráfico para ver si el router "gasta" las IPs del pool.

1. Generar tráfico:

- Ve a la **PC1** y abre el navegador web. Intenta entrar a la IP del servidor externo (o haz un ping).
- Haz lo mismo desde la **L1** (Laptop) rápidamente.

2. Mirar la tabla NAT en R2: Vuelve a la consola de R2 y ejecuta: R2# show ip nat translations

Lo que deberías ver: A diferencia del NAT estático, aquí verás entradas que se crean dinámicamente.

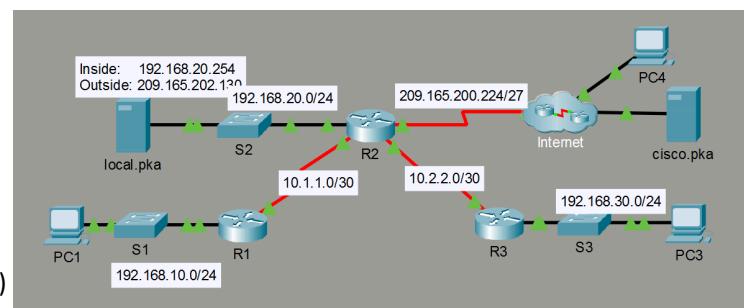
- **Inside Local:** Será la 172.16.x.x (IP privada de tu PC).
- **Inside Global:** Será una de las IPs del pool (ej. 209.165.76.197).

Si esperas un tiempo sin hacer nada (timeout) y vuelves a ejecutar el comando, la tabla estará vacía. ¡Las IPs han vuelto al pool para que las use otro!

EJERCICIO 4 NAT - SOBRECARGA

Parte 1: Configurar NAT Dinámica con PAT (Sobrecarga)

El objetivo aquí es que todas las redes internas (10, 20 y 30) salgan a internet compartiendo la IP .129 usando puertos diferentes.



Paso 1: Configurar la ACL con Nombre (R2NAT)

```
R2(config)# ip access-list standard R2NAT          //Crea una lista llamada R2NAT para definir quién tiene  
permiso para salir  
R2(config-std-nacl)# permit 192.168.10.0 0.0.0.255 // permit [ ip ] [ máscara ]  
R2(config-std-nacl)# permit 192.168.20.0 0.0.0.255  
R2(config-std-nacl)# permit 192.168.30.0 0.0.0.255  
R2(config-std-nacl)# exit
```

Paso 2: Configurar el Pool (R2POOL)

Creamos el pool llamado "R2POOL". Ojo aquí: como solo podemos usar la primera dirección (.129), el inicio y el fin son el mismo número.

- **Red:** 209.165.202.128/30
- **Desglose:**
 - .128 = Dirección de Red (No usable)
 - .129 = Primera IP usable (Esta es la que pide el paso 1)
 - .130 = Segunda IP usable (Esta la guardamos para la estática luego)
 - .131 = Broadcast (No usable)

```
R2(config)# ip nat pool R2POOL 209.165.202.129 209.165.202.129 netmask 255.255.255.252
```

```
//R2(config)#ip nat pool R2POOL 209.165.202.129 209.165.202.129 netmask
```

255.255.255.252 -> la primera dirección del espacio de direcciones de 209.165.202.128/30, es decir, la 129, porque es 128+1.

Paso 3: Asociar la ACL con el Pool + OVERLOAD

Este es el comando más importante. La palabra clave overload activa el PAT. Sin ella, solo un usuario podría navegar.

```
R2(config)# ip nat inside source list R2NAT pool R2POOL overload
```

Paso 4: Configurar las Interfaces

- Serial 0/1/0 (o la que va a Internet): Outside.
- Serial 0/0/0 y Serial 0/0/1 (o las que van a las redes internas): Inside.
- Si hay una Gigabit conectada a una LAN local, también es Inside.

```
R2>enable  
R2#configure terminal  
R2(config)#interface s0/1/0  
R2(config-if)#ip nat outside -> Internet  
R2(config-if)#exit  
R2(config)#interface s0/0/0  
R2(config-if)#ip nat inside  
R2(config-if)#exit  
R2(config)#interface s0/0/1  
R2(config-if)#ip nat inside  
R2(config-if)#exit  
R2(config)#interface fa0/0  
R2(config-if)#ip nat inside  
R2(config-if)#end  
R2#copy running-config startup-config
```

Parte 2: Configurar NAT Estática

Ahora vamos a exponer el servidor local.pka (que está en la red interna) hacia internet usando la segunda IP pública disponible (.130).

! Asumiendo que la IP del servidor local.pka es 192.168.20.254

```
R2(config)# ip nat inside source static 192.168.20.254 209.165.202.130
```

Parte 3: Verificar

- Generar tráfico:
 - Desde PC1 o PC3, abre el navegador y entra a cisco.pka (simula internet). Debería funcionar.
 - Desde PC4 (o un pc externo si lo hay), intenta entrar a local.pka o pon la IP pública 209.165.202.130 en el navegador.

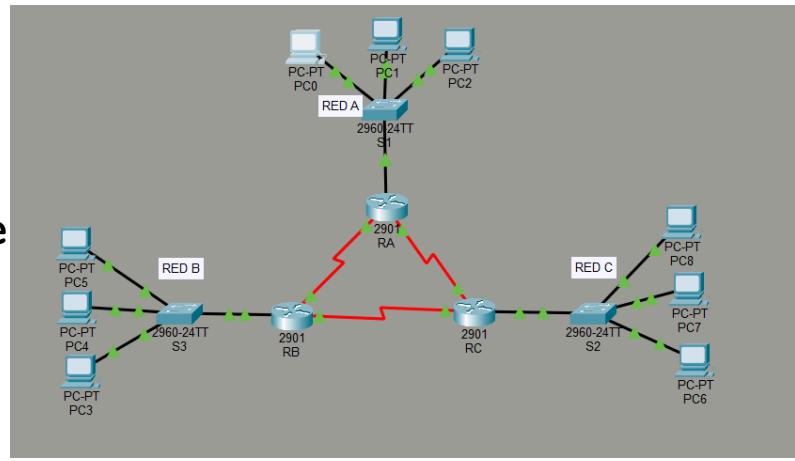
- Ver las tripas del router: Vuelve a R2 y ejecuta: R2# show ip nat translations

PRÁCTICA 5

EJERCICIO 1 Planificación subredes

Parte 1. Diseño del Esquema de Direcciones IPv4 (VLSM)

Objetivo: Dividir la red 192.168.1.0/24 en subredes eficientes para LANs (4 dispositivos) y Enlaces WAN (2 dispositivos).



Lógica de Subneteo (VLSM):

- **Redes LAN (A, B, C):** Necesitamos 4 IPs (1 Router + 3 PCs).
 - Fórmula: $2^n - 2 \geq 4$
 - $n = 3$ bits de host ($2^3 = 8$ direcciones totales).
 - Máscara: $/32 - 3 = /29$ (255.255.255.248).
- **Enlaces WAN (RA-RB, RA-RC, RB-RC):** Necesitamos 2 IPs (Router a Router).
 - Fórmula: $2^n - 2 \geq 2$
 - $n = 2$ bits de host ($2^2 = 4$ direcciones totales).
 - Máscara: $/32 - 2 = /30$ (255.255.255.252).

Tabla de Asignación IPv4

| Descripción | Subred (ID) | Máscara | Primer IP (Router) | Rango Host (PCs/Extremos) | Broadcast |
|--------------|--------------|---------|--------------------|---------------------------|-----------|
| Red A (LAN) | 192.168.1.0 | /29 | RA G0/0: .1 | .2, .3, .4 | .7 |
| Red B (LAN) | 192.168.1.8 | /29 | RB G0/0: .9 | .10, .11, .12 | .15 |
| Red C (LAN) | 192.168.1.16 | /29 | RC G0/0: .17 | .18, .19, .20 | .23 |
| Enlace RA-RB | 192.168.1.24 | /30 | RA S0/0/0: .25 | RB S0/0/0: .26 | .27 |

| | | | | | |
|--------------|--------------|-----|-------------------|----------------|-----|
| Enlace RA-RC | 192.168.1.28 | /30 | RA S0/0/1: .29 | RC S0/0/0: .30 | .31 |
| Enlace RB-RC | 192.168.1.32 | /30 | RB S0/0/1: .33 | RC S0/0/1: .34 | .35 |

Tabla de Asignación IPv6

| Descripción | Subred IPv6 | Dirección Interface Router | Dirección Gateway PCs | Rango PCs (Sufijo) |
|--------------|----------------------|----------------------------|-----------------------|--------------------|
| Red A | 2001:db8:acad:1::/64 | ...:1::1 | fe80::1 | ::2, ::3, ::4 |
| Red B | 2001:db8:acad:2::/64 | ...:2::1 | fe80::1 | ::2, ::3, ::4 |
| Red C | 2001:db8:acad:3::/64 | ...:3::1 | fe80::1 | ::2, ::3, ::4 |
| Enlace RA-RB | 2001:db8:acad:4::/64 | RA: ...:4::1 | N/A | RB: ...:4::2 |
| Enlace RA-RC | 2001:db8:acad:5::/64 | RA: ...:5::1 | N/A | RC: ...:5::2 |
| Enlace RB-RC | 2001:db8:acad:6::/64 | RB: ...:6::1 | N/A | RC: ...:6::2 |

Paso 1 Configuración Router A

```
enable
configure terminal
hostname RA
ipv6 unicast-routing //Esencial para que el router mueva tráfico IPv6.
```

```
! LAN A
interface g0/0
description LAN Red A
```

```
ip address 192.168.1.1 255.255.255.248
ipv6 address 2001:db8:acad:1::1/64
ipv6 address fe80::1 link-local //Estandarizamos el Gateway para que sea fácil de recordar en los PCs.
no shutdown
exit
```

```
! WAN hacia RB
interface s0/0/0
description Link a RB
ip address 192.168.1.25 255.255.255.252
ipv6 address 2001:db8:acad:4::1/64
clock rate 128000 //Se aplica en el extremo del cable serial que tiene el reloj (DCE).
no shutdown
exit
```

```
! WAN hacia RC
interface s0/0/1
description Link a RC
ip address 192.168.1.29 255.255.255.252
ipv6 address 2001:db8:acad:5::1/64
clock rate 128000
no shutdown
exit
```

Paso 2 Configuración Router B

```
enable
configure terminal
hostname RB
ipv6 unicast-routing
```

```
! LAN B
interface g0/0
description LAN Red B
ip address 192.168.1.9 255.255.255.248
ipv6 address 2001:db8:acad:2::1/64
ipv6 address fe80::1 link-local
no shutdown
exit
```

```
! WAN hacia RA
interface s0/0/0
description Link a RA
ip address 192.168.1.26 255.255.255.252
ipv6 address 2001:db8:acad:4::2/64
no shutdown
exit
```

```

! WAN hacia RC
interface s0/0/1
description Link a RC
ip address 192.168.1.33 255.255.255.252
ipv6 address 2001:db8:acad:6::1/64
clock rate 128000
no shutdown
exit

```

Paso 3 Configuración Router C

```

enable
configure terminal
hostname RC
ipv6 unicast-routing

```

```

! LAN C
interface g0/0
description LAN Red C
ip address 192.168.1.17 255.255.255.248
ipv6 address 2001:db8:acad:3::1/64
ipv6 address fe80::1 link-local
no shutdown
exit

```

```

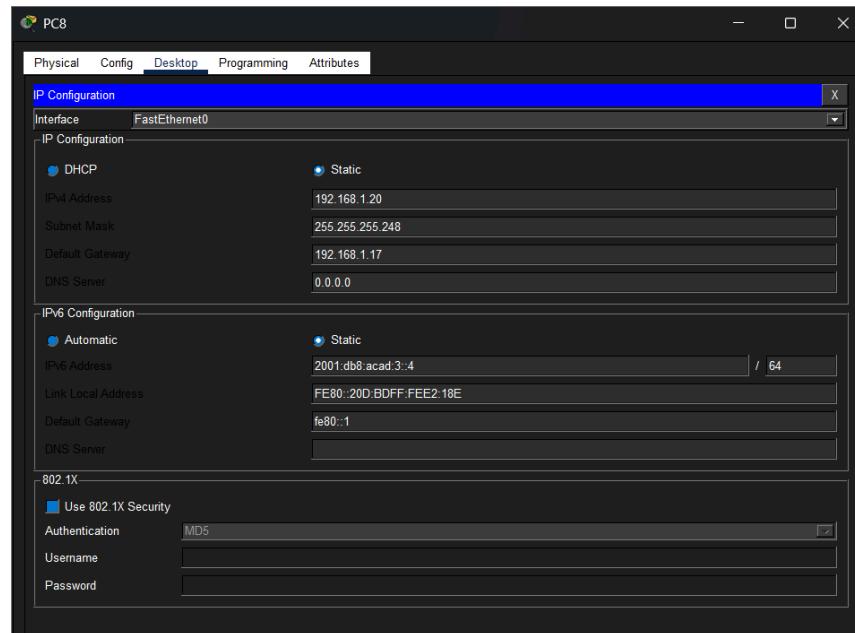
! WAN hacia RA
interface s0/0/0
description Link a RA
ip address 192.168.1.30 255.255.255.252
ipv6 address 2001:db8:acad:5::2/64
no shutdown
exit

```

```

! WAN hacia RB
interface s0/0/1
description Link a RB
ip address 192.168.1.34 255.255.255.252
ipv6 address 2001:db8:acad:6::2/64
no shutdown
exit

```



Paso 4 Configuración de los PCs

| PC | Red (Ubicación) | Dirección IPv4 | Máscara de Subred | Gateway IPv4 | Dirección IPv6 (Prefijo /64) | Gateway IPv6 |
|-----|--------------------|-------------------|----------------------|--------------|---------------------------------|-----------------|
| PC0 | Red A (RA) | 192.168.1.2 | 255.255.255.248 | 192.168.1.1 | 2001:db8:acad:1::2 | fe80::1 |
| PC1 | Red A (RA) | 192.168.1.3 | 255.255.255.248 | 192.168.1.1 | 2001:db8:acad:1::3 | fe80::1 |
| PC2 | Red A (RA) | 192.168.1.4 | 255.255.255.248 | 192.168.1.1 | 2001:db8:acad:1::4 | fe80::1 |
| PC3 | Red B (RB) | 192.168.1.10 | 255.255.255.248 | 192.168.1.9 | 2001:db8:acad:2::2 | fe80::1 |
| PC4 | Red B (RB) | 192.168.1.11 | 255.255.255.248 | 192.168.1.9 | 2001:db8:acad:2::3 | fe80::1 |
| PC5 | Red B (RB) | 192.168.1.12 | 255.255.255.248 | 192.168.1.9 | 2001:db8:acad:2::4 | fe80::1 |
| PC6 | Red C (RC) | 192.168.1.18 | 255.255.255.248 | 192.168.1.17 | 2001:db8:acad:3::2 | fe80::1 |
| PC7 | Red C (RC) | 192.168.1.19 | 255.255.255.248 | 192.168.1.17 | 2001:db8:acad:3::3 | fe80::1 |
| PC8 | Red C (RC) | 192.168.1.20 | 255.255.255.248 | 192.168.1.17 | 2001:db8:acad:3::4 | fe80::1 |

Parte 2. Verificación de Conectividad

Para comprobar que el ejercicio 1 está completo:

1. Ping Local (LAN): Desde PC0, hacer ping a PC1 (192.168.1.3).
 - *Resultado esperado:* Successful.
2. Ping al Gateway: Desde PC0, hacer ping a RA (192.168.1.1).
 - *Resultado esperado:* Successful.
3. Ping Remoto (WAN): Desde PC0, hacer ping a PC3 (192.168.1.10).
 - *Resultado esperado:* Failed / Unreachable.
 - *Razón:* Los routers tienen las direcciones puestas, pero aún no saben "cómo llegar" a las otras redes (falta el protocolo de enrutamiento del Ejercicio 2).

EJERCICIO 2 Enrutamiento estático

Parte:1. Enrutamiento Estático IPv4 y IPv6

Lógica del Diseño: Para cumplir con el requisito de "camino con menor número de saltos", configuraremos las rutas de la siguiente manera:

- Router A: Debe saber llegar a la LAN B (saltando a RB) y a la LAN C (saltando a RC).
- Router B: Debe saber llegar a la LAN A (saltando a RA) y a la LAN C (saltando a RC).
- Router C: Debe saber llegar a la LAN A (saltando a RA) y a la LAN B (saltando a RB).

(Nota: Usamos la máscara 255.255.255.248 porque las LANs son /29).

Paso 1 Configuración RA

RA conoce directamente sus propias redes (LAN A, Enlace a B, Enlace a C). Necesitamos enseñarle a llegar a:

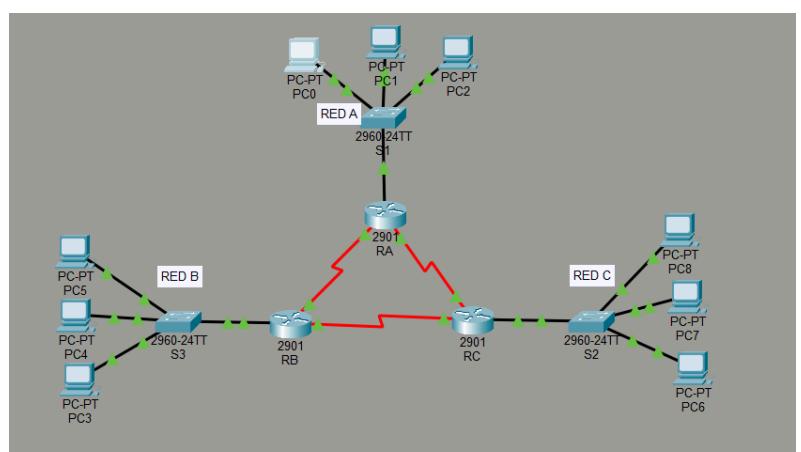
1. **LAN B:** Saltando directamente al Router B.
2. **LAN C:** Saltando directamente al Router C.
3. **Enlace WAN RB-RC:** Saltando al Router B (o C).
4. **Ruta por defecto:** Para tráfico desconocido.

! --- Configuración Global ---

```
enable  
configure terminal  
ipv6 unicast-routing
```

! --- Rutas Estáticas IPv4 ---

```
! Hacia LAN B (vía IP de RB)  
ip route 192.168.1.8 255.255.255.248 192.168.1.26  
! Hacia LAN C (vía IP de RC)  
ip route 192.168.1.16 255.255.255.248 192.168.1.30  
! Hacia Enlace WAN RB-RC (vía IP de RB)  
ip route 192.168.1.32 255.255.255.252 192.168.1.26
```



```
! Ruta por defecto (Gateway of Last Resort)
ip route 0.0.0.0 0.0.0.0 192.168.1.26

! --- Rutas Estáticas IPv6 ---
! Hacia LAN B (vía IP IPv6 de RB)
ipv6 route 2001:db8:acad:2::/64 2001:db8:acad:4::2
! Hacia LAN C (vía IP IPv6 de RC)
ipv6 route 2001:db8:acad:3::/64 2001:db8:acad:5::2
! Hacia Enlace WAN RB-RC (vía IP IPv6 de RB)
ipv6 route 2001:db8:acad:6::/64 2001:db8:acad:4::2
! Ruta por defecto IPv6
ipv6 route ::/0 2001:db8:acad:4::2
```

```
exit
```

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

Paso 2 Configuración RB

RB conoce sus redes. Necesitamos enseñarle a llegar a:

- LAN A: Saltando directamente al Router A.
- LAN C: Saltando directamente al Router C.
- Enlace WAN RA-RC: Saltando al Router A.
- Ruta por defecto: Para tráfico desconocido.

```
! --- Configuración Global ---
```

```
enable
configure terminal
ipv6 unicast-routing
```

```
! --- Rutas Estáticas IPv4 ---
```

```
! Hacia LAN A (vía IP de RA)
ip route 192.168.1.0 255.255.255.248 192.168.1.25
! Hacia LAN C (vía IP de RC)
ip route 192.168.1.16 255.255.255.248 192.168.1.34
! Hacia Enlace WAN RA-RC (vía IP de RA)
ip route 192.168.1.28 255.255.255.252 192.168.1.25
! Ruta por defecto
ip route 0.0.0.0 0.0.0.0 192.168.1.25
```

```
! --- Rutas Estáticas IPv6 ---
```

```
! Hacia LAN A (vía IP IPv6 de RA)
ipv6 route 2001:db8:acad:1::/64 2001:db8:acad:4::1
! Hacia LAN C (vía IP IPv6 de RC)
ipv6 route 2001:db8:acad:3::/64 2001:db8:acad:6::2
! Hacia Enlace WAN RA-RC (vía IP IPv6 de RA)
ipv6 route 2001:db8:acad:5::/64 2001:db8:acad:4::1
! Ruta por defecto IPv6
ipv6 route ::/0 2001:db8:acad:4::1
```

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

Paso 3 Configuración RC

RC conoce sus redes. Necesitamos enseñarle a llegar a:

1. **LAN A:** Saltando directamente al Router A.
2. **LAN B:** Saltando directamente al Router B.
3. **Enlace WAN RA-RB:** Saltando al Router A.
4. **Ruta por defecto:** Para tráfico desconocido.

! --- Configuración Global ---

```
enable
configure terminal
ipv6 unicast-routing
```

! --- Rutas Estáticas IPv4 ---

```
! Hacia LAN A (vía IP de RA)
ip route 192.168.1.0 255.255.255.248 192.168.1.29
! Hacia LAN B (vía IP de RB) - CAMINO MÁS CORTO
ip route 192.168.1.8 255.255.255.248 192.168.1.33
! Hacia Enlace WAN RA-RB (vía IP de RA)
ip route 192.168.1.24 255.255.255.252 192.168.1.29
! Ruta por defecto
ip route 0.0.0.0 0.0.0.0 192.168.1.29
```

! --- Rutas Estáticas IPv6 ---

```
! Hacia LAN A (vía IP IPv6 de RA)
ipv6 route 2001:db8:acad:1::/64 2001:db8:acad:5::1
! Hacia LAN B (vía IP IPv6 de RB) - CAMINO MÁS CORTO
ipv6 route 2001:db8:acad:2::/64 2001:db8:acad:6::1
! Hacia Enlace WAN RA-RB (vía IP IPv6 de RA)
ipv6 route 2001:db8:acad:4::/64 2001:db8:acad:5::1
! Ruta por defecto IPv6
ipv6 route ::/0 2001:db8:acad:5::1
```

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

Parte 2. Verificación de Conectividad

Para validar que el ejercicio está correcto al 100%:

- Ping PC0 (Red A) -> PC3 (Red B): 192.168.1.10 (Debe ser Successful).
- Ping PC0 (Red A) -> PC6 (Red C): 192.168.1.18 (Debe ser Successful).
- Ping PC3 (Red B) -> PC6 (Red C): 192.168.1.18 (Debe ser Successful).
- Verificación IPv6: Repetir los pings usando las direcciones IPv6 (ej: ping 2001:db8:acad:2::2).

EJERCICIO 3 Enrutamiento EIGRP

Objetivo: Configurar el protocolo de enrutamiento dinámico EIGRP para que los routers aprendan automáticamente las rutas de la red. Datos clave:

- **Sistema Autónomo (AS):** Usaremos el 1 (debe ser el mismo en todos los routers).
- **Wildcards:** EIGRP usa "máscaras inversas" (wildcards).
 - Para la LAN (/29): 0.0.0.7
 - Para la WAN (/30): 0.0.0.3

1. Configuración del Router A (RA)

- Router ID: 1.1.1.1
- Redes a anunciar: LAN A, Enlace a RB, Enlace a RC.
- Interfaz Pasiva: La G0/0 (LAN), porque ahí no hay routers escuchando, solo PCs, y no queremos enviar "ruido" EIGRP hacia ellos.

enable

configure terminal

! Entramos al modo de configuración EIGRP con el Sistema Autónomo 1

router eigrp 1

! Asignamos el ID del Router (como su DNI)

eigrp router-id 1.1.1.1

! Anunciamos las redes conectadas DIRECTAMENTE a RA (usando Wildcards)

! Red LAN A

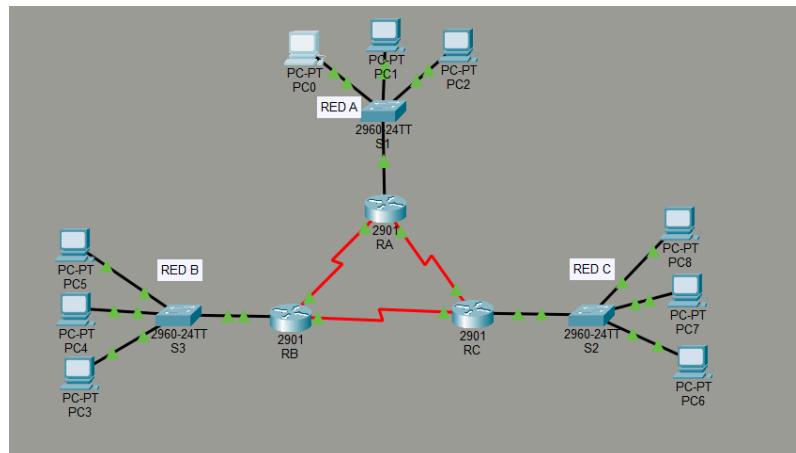
network 192.168.1.0 0.0.0.7

! Enlace WAN hacia RB

network 192.168.1.24 0.0.0.3

! Enlace WAN hacia RC

network 192.168.1.28 0.0.0.3



! Configuramos la interfaz LAN como pasiva (seguridad y eficiencia)

passive-interface g0/0

! Desactivamos el resumen automático (vital para que no mezcle subredes)

no auto-summary

exit

copy running-config startup-config

2. Configuración del Router B (RB)

- Router ID: 2.2.2.2
- Redes a anunciar: LAN B, Enlace a RA, Enlace a RC.

enable

configure terminal

router eigrp 1

eigrp router-id 2.2.2.2

```
! Red LAN B
network 192.168.1.8 0.0.0.7
! Enlace WAN hacia RA
network 192.168.1.24 0.0.0.3
! Enlace WAN hacia RC
network 192.168.1.32 0.0.0.3
! Interfaz pasiva (hacia los PCs)
passive-interface g0/0
! Sin resumen automático
no auto-summary
exit
copy running-config startup-config
```

3. Configuración del Router C (RC)

- Router ID: 3.3.3.3
- Redes a anunciar: LAN C, Enlace a RA, Enlace a RB.

```
enable
configure terminal
router eigrp 1
eigrp router-id 3.3.3.3
! Red LAN C
network 192.168.1.16 0.0.0.7
! Enlace WAN hacia RA
network 192.168.1.28 0.0.0.3
! Enlace WAN hacia RB
network 192.168.1.32 0.0.0.3
! Interfaz pasiva
passive-interface g0/0
! Sin resumen automático
no auto-summary

exit
copy running-config startup-config
```

4. Verificación y Análisis (Para tu documento)

Una vez configurado, verás mensajes en la consola diciendo DUAL-5-NBRCHANGE: IP-EIGRP... New Adjacency.
Eso significa que los routers se han "hecho amigos".

Para analizar que todo está correcto, usa estos comandos de verificación (y haz capturas para tu doc):

1. **Ver la tabla de vecinos (¿Con quién hablo?):** show ip eigrp neighbors
Deberías ver 2 vecinos en cada router.
2. **Ver la tabla de rutas (¿Qué he aprendido?):** show ip route eigrp
Deberías ver las redes de los otros routers marcadas con una letra "D" (de DUAL, el algoritmo de EIGRP).

3. **Prueba de conectividad:** Igual que antes, haz un Ping de PC0 (Red A) a PC6 (Red C). ¡Debería funcionar automáticamente sin haber puesto ni una sola ruta estática!

EJERCICIO 4 Enrutamiento OSPF

Router A (RA) - OSPF

```
! OSPFv2 (IPv4)
router ospf 1
router-id 1.1.1.1
network 192.168.1.0 0.0.0.7 area 0
network 192.168.1.24 0.0.0.3 area 0
network 192.168.1.28 0.0.0.3 area 0
passive-interface g0/0
exit
```

```
! OSPFv3 (IPv6)
ipv6 router ospf 10
router-id 1.1.1.1
passive-interface g0/0
exit
```

```
interface g0/0
ipv6 address 2001:db8:acad:1::1/64
ipv6 ospf 10 area 0
exit
interface s0/0/0
ipv6 address 2001:db8:acad:4::1/64
ipv6 ospf 10 area 0
exit
interface s0/0/1
ipv6 address 2001:db8:acad:5::1/64
ipv6 ospf 10 area 0
exit
```

Router B (RB) - OSPF

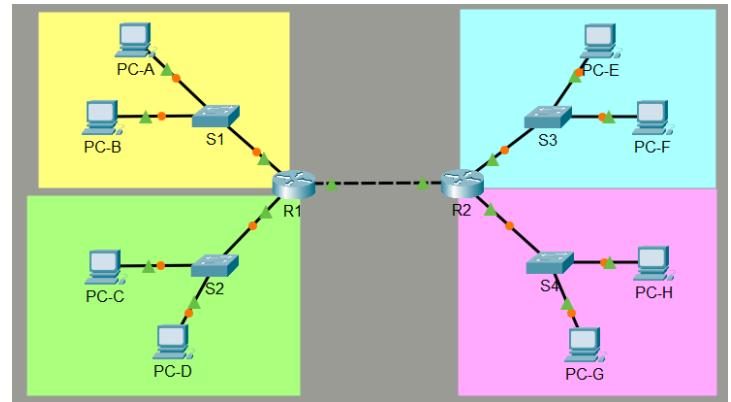
```
! OSPFv2 (IPv4)
router ospf 1
router-id 2.2.2.2
network 192.168.1.8 0.0.0.7 area 0
network 192.168.1.24 0.0.0.3 area 0
network 192.168.1.32 0.0.0.3 area 0
passive-interface g0/0
exit
```

```
! OSPFv3 (IPv6)
ipv6 router ospf 10
router-id 2.2.2.2
passive-interface g0/0
exit
```

```

interface g0/0
  ipv6 address 2001:db8:acad:2::1/64
  ipv6 ospf 10 area 0
exit
interface s0/0/0
  ipv6 address 2001:db8:acad:4::2/64
  ipv6 ospf 10 area 0
exit
interface s0/0/1
  ipv6 address 2001:db8:acad:6::1/64
  ipv6 ospf 10 area 0
exit

```



Router C (RC) - OSPF

```

! OSPFv2 (IPv4)
router ospf 1
  router-id 3.3.3.3
  network 192.168.1.16 0.0.0.7 area 0
  network 192.168.1.28 0.0.0.3 area 0
  network 192.168.1.32 0.0.0.3 area 0
  passive-interface g0/0
exit

```

```

! OSPFv3 (IPv6)
ipv6 router ospf 10
  router-id 3.3.3.3
  passive-interface g0/0
exit

```

```

interface g0/0
  ipv6 address 2001:db8:acad:3::1/64
  ipv6 ospf 10 area 0
exit
interface s0/0/0
  ipv6 address 2001:db8:acad:5::2/64
  ipv6 ospf 10 area 0
exit
interface s0/0/1
  ipv6 address 2001:db8:acad:6::2/64
  ipv6 ospf 10 area 0
exit

```

PRÁCTICA 6

Ejercicio 1 - listas de acceso IPv4 estándar

Objetivo: Restringir el tráfico de red y el acceso de gestión remota utilizando Listas de Control de Acceso (ACL). Estándar numeradas y nombradas.

1. Configuración del Router R2 (Filtrado de Tráfico LAN)

```
! --- ACL 10: FILTRO PARA LAN ROSA ---  
! Permitir PC-C  
access-list 10 remark ACL_TO_PINK_LAN  
access-list 10 permit host 192.168.2.50  
! Permitir mitad inferior de LAN Amarilla (0-127)  
access-list 10 permit 192.168.1.0 0.0.0.127  
! Permitir LAN Azul  
access-list 10 permit 172.16.1.0 0.0.0.255
```

```
! Aplicar en G0/1 hacia afuera (outbound)  
interface GigabitEthernet0/1  
ip access-group 10 out  
exit
```

```
! --- ACL 20: FILTRO PARA LAN AZUL ---  
! Permitir PC-A (Excepción)  
access-list 20 remark ACL_TO_BLUE_LAN  
access-list 20 permit host 192.168.1.100  
! Denegar resto de LAN Amarilla  
access-list 20 deny 192.168.1.0 0.0.0.255  
! Permitir todo lo demás  
access-list 20 permit any
```

```
! Aplicar en G0/0 hacia afuera (outbound)  
interface GigabitEthernet0/0  
ip access-group 20 out  
exit
```

2. Configuración del Router R1 (Seguridad Acceso Remoto)

```
! Crear ACL Estándar con nombre  
ip access-list standard ADMIN_VTY  
permit host 192.168.2.50  
exit
```

```
! Aplicar a las líneas virtuales (VTY 0-15) en dirección de entrada  
line vty 0 4  
access-class ADMIN_VTY in  
exit  
line vty 5 15  
access-class ADMIN_VTY in
```

exit

3. Pruebas de Verificación (Matriz de Resultados)

| Origen | Destino (IP) | Protocolo | Resultado Esperado | Razón Técnica |
|--------|---------------------|-------------|--------------------|--|
| PC-A | LAN Rosa (.200) | ICMP (Ping) | Éxito | Permitido por permit 192.168.1.0 0.0.0.127 en ACL 10. |
| PC-B | LAN Rosa (.200) | ICMP (Ping) | Fallo | IP .150 fuera del rango .0-.127 (Deny implícito ACL 10). |
| PC-C | LAN Rosa (.200) | ICMP (Ping) | Éxito | Permitido explícitamente host en ACL 10. |
| PC-A | LAN Azul (.10) | ICMP (Ping) | Éxito | Excepción permitida host en ACL 20. |
| PC-B | LAN Azul (.10) | ICMP (Ping) | Fallo | Denegado por deny 192.168.1.0 en ACL 20. |
| PC-C | LAN Azul (.10) | ICMP (Ping) | Éxito | Permitido por permit any en ACL 20. |
| PC-C | R1 (192.168.2.1) | SSH | Éxito | Permitido en ADMIN_VTY. |
| PC-A | R1 (192.168.1.1) | SSH | Fallo | Denegado implícitamente en ADMIN_VTY. |

Ejercicio 2: Configuración de ACLs Extendidas Numeradas

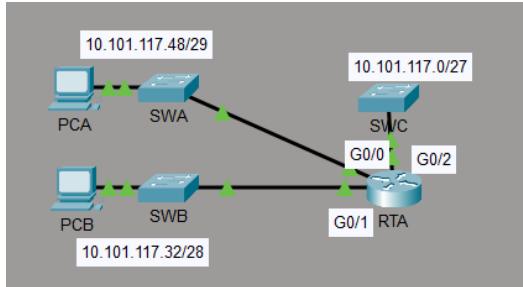
Objetivo: Implementar una política de seguridad utilizando una Lista de Control de Acceso (ACL) Extendida para filtrar el tráfico hacia una red específica, permitiendo servicios de gestión (Telnet) solo desde orígenes autorizados y permitiendo el diagnóstico (ICMP) global.

1. Análisis de Requisitos y Cálculos

Se debe configurar la ACL 199 en el router RTA.

Política de Seguridad:

1. Permitir Telnet desde la red 10.101.117.32/28 hacia la red 10.101.117.0/27.
2. Permitir ICMP desde cualquier origen a cualquier destino.
3. Denegar todo lo demás hacia la red destino.



Cálculo de Wildcards:

- Origen (.32/28): Máscara 255.255.255.240. Wildcard: $255.255.255.255 - 255.255.255.240 = 0.0.0.15$
- Destino (.0/27): Máscara 255.255.255.224. Wildcard: $255.255.255.255 - 255.255.255.224 = 0.0.0.31$

Ubicación de la ACL:

Aunque las ACLs extendidas suelen colocarse cerca del origen, en este escenario específico se solicita aplicarla en la interfaz GigabitEthernet0/2 en dirección de salida (out), para filtrar el tráfico justo antes de que entre a la red destino (10.101.117.0/27).

2. Configuración del Router RTA

! --- Paso 1: Definir la ACL Extendida 199 ---

! Regla 1: Permitir Telnet (puerto 23) desde la red .32 a la .0

! Sintaxis: access-list <num> permit tcp <origen> <wildcard> <destino> <wildcard> eq <puerto>
access-list 199 permit tcp 10.101.117.32 0.0.0.15 10.101.117.0 0.0.0.31 eq telnet

! Regla 2: Permitir ICMP (Ping) desde cualquier lugar a cualquier lugar

access-list 199 permit icmp any any

! Regla 3: Denegación implícita

! (No es necesario escribirla, está oculta al final: deny ip any any)

! --- Paso 2: Aplicar la ACL a la Interfaz ---

! Seleccionamos la interfaz que conecta con la red destino (.0/27)

interface GigabitEthernet0/2

description Conexion hacia SWC y Red Destino

! Aplicamos la lista en dirección de SALIDA (hacia la red)

ip access-group 199 out

exit

3. Matriz de Verificación y Pruebas

Para validar el funcionamiento correcto de la ACL, se deben realizar las siguientes pruebas desde los distintos equipos.

| Origen | Destino | Protocolo | Resultado Esperado | Explicación Técnica |
|-----------|----------|-----------------|--------------------|---|
| PCB (.35) | SWC (.2) | ICMP (Ping) | Éxito | Permitido por la regla permit icmp any any. |
| PCB (.35) | SWC (.2) | TCP/23 (Telnet) | Éxito | Permitido explícitamente: PCB pertenece a la red origen .32/28. |
| PCA (.51) | SWC (.2) | ICMP (Ping) | Éxito | Permitido por la regla permit icmp any any. |
| PCA (.51) | SWC (.2) | TCP/23 (Telnet) | Fallo | Denegado implícitamente. PCA pertenece a la red .48/29, que no coincide con la regla de Telnet. |

4. Reflexión y Análisis (Parte 2 de la Práctica)

Pregunta G: Despues de iniciar sesión en RTA, se intenta acceder al SWC mediante Telnet desde el propio router. ¿Qué ocurre y cómo se podría haber evitado?

Análisis:

- **Lo que ocurre:** El Telnet desde RTA hacia SWC tiene **éxito**, a pesar de que la IP de RTA no está en el rango permitido .32/28.
- **Causa Técnica:** Por diseño en los equipos Cisco, **el tráfico generado localmente por el propio router no se filtra por las ACLs aplicadas en sus interfaces de salida**. La ACL 199 en G0/2 out filtra el tráfico que *atraviesa* el router, no el que *nace* en él.
- **Solución:** Para evitar esto y restringir estrictamente el acceso al SWC, la ACL debería haberse aplicado en las **líneas VTY (acceso virtual) del SWC** (destino final) utilizando el comando access-class, en lugar de en la interfaz física del router. Alternativamente, configurar una ACL de "Control Plane" en RTA, aunque es una configuración más avanzada.

Ejercicio 3 - listas de acceso IPv4 extendida con nombre

Nombre de la ACL: ACL (según pide el enunciado). Ubicación: Router RT1. Interfaz: G0/0 (La puerta de enlace de la LAN). Dirección: in (Entrada). ¿Por qué "in"? Porque el tráfico viene de los PCs (PC1, PC2, PC3) y entra al router por la interfaz G0/0 para salir hacia Internet. Es más eficiente bloquearlo nada más entrar.

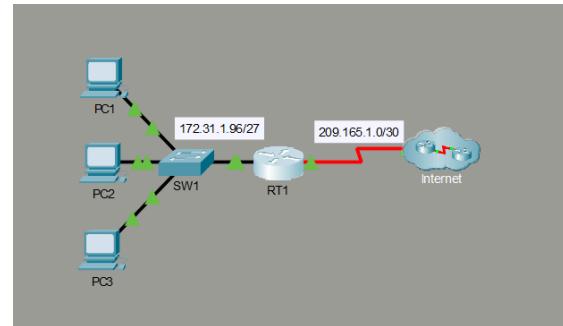
Lógica de Reglas (Orden Secuencial)

- PC1: Bloquear HTTP (80) y HTTPS (443) hacia Server1 y Server2.
- PC2: Bloquear FTP (21) hacia Server1 y Server2.
- PC3: Bloquear ICMP (Ping) hacia Server1 y Server2.
- Resto: Permitir todo (ip any any).

Configuración en Router RT1

```
enable
```

```
configure terminal
```



! --- Paso 1: Crear la ACL Extendida llamada "ACL" ---

```
ip access-list extended ACL
```

! --- Reglas para PC1 (Bloquear Web) ---

! Denegar HTTP (80) y HTTPS (443) hacia Server1 (64.101.255.254)

```
deny tcp host 172.31.1.101 host 64.101.255.254 eq 80
```

```
deny tcp host 172.31.1.101 host 64.101.255.254 eq 443
```

! Denegar HTTP (80) y HTTPS (443) hacia Server2 (64.103.255.254)

```
deny tcp host 172.31.1.101 host 64.103.255.254 eq 80
```

```
deny tcp host 172.31.1.101 host 64.103.255.254 eq 443
```

! --- Reglas para PC2 (Bloquear FTP) ---

! Denegar FTP (21) hacia Server1 y Server2

```
deny tcp host 172.31.1.102 host 64.101.255.254 eq 21
```

```
deny tcp host 172.31.1.102 host 64.103.255.254 eq 21
```

! --- Reglas para PC3 (Bloquear Ping) ---

! Denegar ICMP hacia Server1 y Server2

```
deny icmp host 172.31.1.103 host 64.101.255.254
```

```
deny icmp host 172.31.1.103 host 64.103.255.254
```

! --- Paso 4: Permitir el resto del tráfico ---

! Vital: Si no pones esto, se bloquea TODO (deny implícito)

```
permit ip any any
```

```
exit
```

! --- Parte 2: Aplicar la ACL a la interfaz ---

! Interfaz G0/0 (Gateway de la LAN)

```
interface GigabitEthernet0/0
```

```
ip access-group ACL in
```

```
exit
```

```
end
```

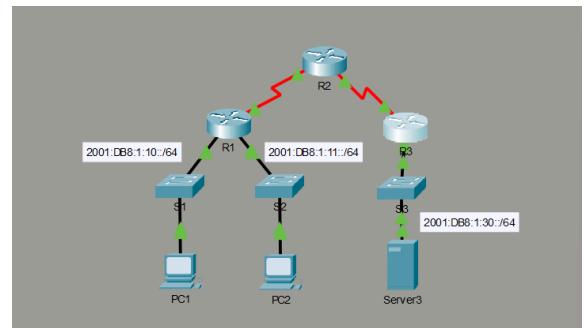
```
copy running-config startup-config
```

Ejercicio 4 - listas de acceso IPv6

Objetivo: Proteger la red contra ataques de Denegación de Servicio (DoS) bloqueando tráfico HTTP/HTTPS específico en origen y tráfico ICMP (Ping) masivo en destino.

1. Configuración del Router R1 (ACL BLOCK_HTTP)

```
enable  
configure terminal
```



! --- Paso 1: Crear la ACL IPv6 ---

```
ipv6 access-list BLOCK_HTTP  
! Bloquear tráfico Web (80) hacia Server3  
deny tcp any host 2001:DB8:1:30::30 eq www  
! Bloquear tráfico Seguro (443) hacia Server3  
deny tcp any host 2001:DB8:1:30::30 eq 443  
! Permitir el resto del tráfico (VITAL)  
permit ipv6 any any  
exit
```

! --- Paso 2: Aplicar a la interfaz (Origen) ---

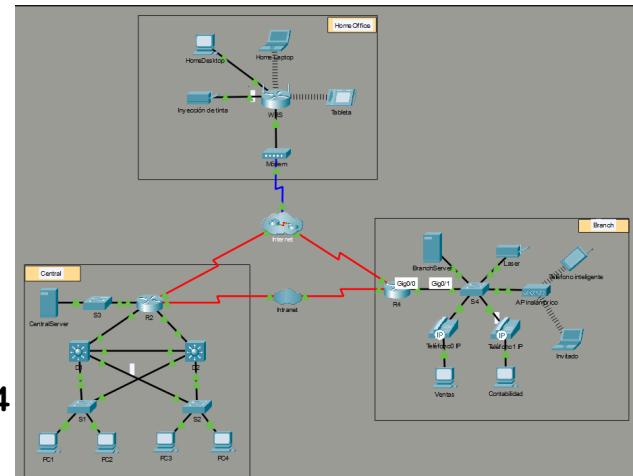
! (Suponiendo que G0/1 es la que conecta con la red 11::0, verifícalo en tu mapa)

```
interface GigabitEthernet0/1  
ipv6 traffic-filter BLOCK_HTTP in  
exit  
  
end  
copy running-config startup-config
```

PRÁCTICA 7

Ejercicio 1: DHCP - DNS

Objetivo: Configurar una red doméstica para que asigne IPs automáticamente (DHCP) y configurar un servidor DNS en internet para resolver nombres de dominio a direcciones IP.



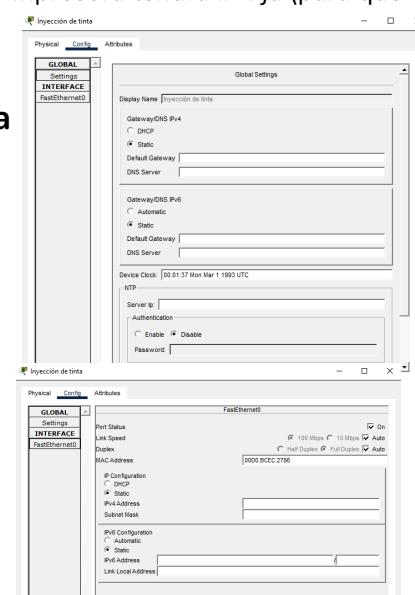
Parte 1: Configuración de Direccionamiento IPv4 (Estático y DHCP)

En esta parte configuraremos la red local "doméstica" (192.168.0.0/24). La impresora tendrá IP fija (para que siempre la encontremos) y el resto irá por DHCP.

Paso 1: Configurar la Impresora (Inkjet) con IP Estática

Las impresoras, servidores y routers suelen llevar IP estática.

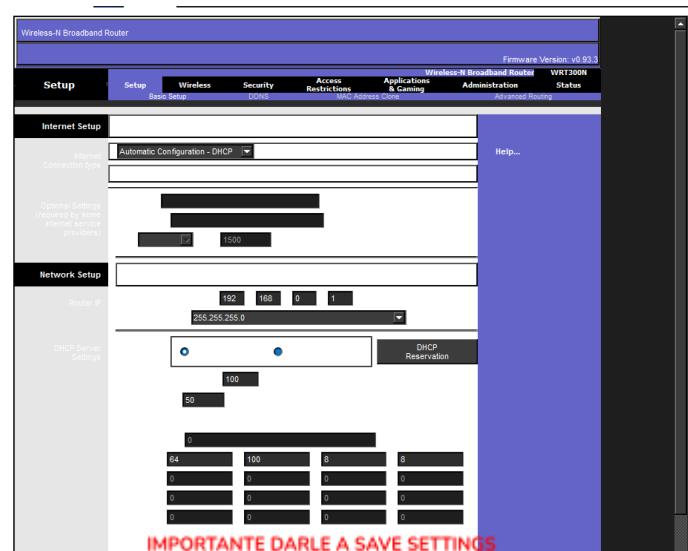
1. Haz clic en el dispositivo **Inkjet**.
2. Ve a la pestaña **Config > Global Settings**.
 - o **Gateway:** 192.168.0.1 (Será la IP del router WRS).
 - o **DNS Server:** 64.100.8.8 (El servidor DNS de Internet).
3. Ve a la pestaña **Config > FastEthernet0** (Interfaz de red).
 - o **IP Configuration:** Static.
 - o **IP Address:** 192.168.0.2
 - o **Subnet Mask:** 255.255.255.0

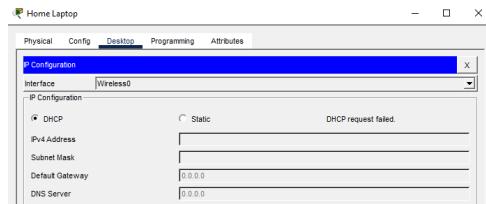


Paso 2: Configurar el Router Inalámbrico (WRS) como Servidor DHCP

El WRS actuará como el "jefe" de la red local, repartiendo IPs a los portátiles y tabletas.

1. Haz clic en **WRS**.
2. Ve a la pestaña **GUI > Setup > Basic Setup**.
3. **Network Setup (Configuración de Red):**
 - o **IP Address:** Cambiar a 192.168.0.1 (Esta es su propia IP y el Gateway de la red).
 - o **Subnet Mask:** 255.255.255.0
 - o **DHCP Server:** Asegúrate de que esté en **Enabled**.
 - o **Static DNS 1:** 64.100.8.8 (Para que los clientes sepan a quién preguntar por los nombres web).
4. **Importante:** Baja al final de la página y haz clic en **Save Settings**. (Si no guardas, al cambiar de pestaña se borra).





Paso 3 y 4: Solicitar IP por DHCP (Clientes)

Ahora vamos a decirle a los equipos que pidan una IP al router WRS.

1. Home Laptop:

- Haz clic en el portátil > Pestaña **Desktop** > **IP Configuration**.
- Cambia de "Static" a **DHCP**.
- *Resultado:* Debería aparecer "DHCP request successful" y asignarle una IP tipo 192.168.0.1xx.

2. Tablet:

- Haz clic en la Tablet > Pestaña **Desktop** > **IP Configuration**.
- Cambia a **DHCP**.
- *Resultado:* "DHCP request successful".

Paso 5: Comprobar acceso Web (Por IP)

Como aún no hemos configurado el DNS, solo podemos navegar usando números.

1. Abre el navegador en **Home Laptop**.
2. URL: 10.10.10.2 (Debería cargar "Central Server").
3. URL: 64.100.200.1 (Debería cargar "Branch Server").
4. *Prueba de fallo:* Si escribes centralserver.pt.pka, dará error "Host Name Unresolved" (porque falta la Parte 2).

Parte 2: Configuración de Registros DNS

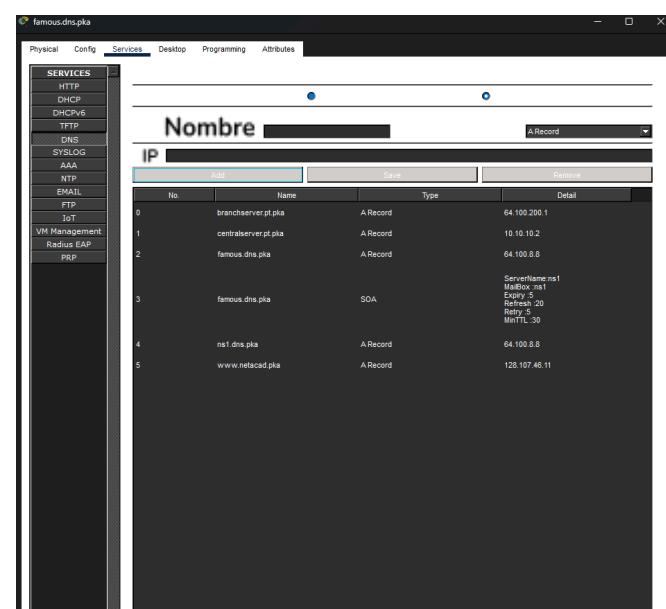
Ahora vamos a hacer la magia: asociar esos nombres (.pka) a las IPs numéricas.

Paso 1: Configurar el Servidor "famous.dns.pka"

Este servidor está en la nube (Internet).

1. Haz clic en la nube **Internet** para entrar dentro del clúster.
2. Haz clic en el servidor **famous.dns.pka**.
3. Ve a la pestaña **Services** > **DNS**.
4. Asegúrate de que el servicio esté **On**.
5. **Añadir Registros (Resource Records):**

- **Registro 1:**
 - **Name:** centralserver.pt.pka
 - **Type:** A Record
 - **Address:** 10.10.10.2
 - Haz clic en **Add**.
- **Registro 2:**
 - **Name:** branchserver.pt.pka
 - **Type:** A Record
 - **Address:** 64.100.200.1
 - Haz clic en **Add**.



6. Cierra la ventana y haz clic en **Back** (o la flecha de volver) para salir de la nube.

Paso 2: Verificación Final (DNS en Acción)

Vamos a comprobar que los clientes ahora pueden "traducir" nombres a IPs.

1. Abre **Home Laptop o Tablet**.

2. **Prueba técnica (Command Prompt):**

- Escribe ipconfig /all. Verifica que el servidor DNS sea 64.100.8.8.
- Haz ping al DNS: ping 64.100.8.8. (Si falla el primero, espera un poco o dale a *Fast Forward Time*).
- Prueba de resolución: nslookup centralserver.pt.pka.
 - Resultado esperado: El servidor famous.dns.pka debe responder con la IP 10.10.10.2.

3. **Prueba de usuario (Navegador Web):**

- Abre el navegador.
- Escribe centralserver.pt.pka -> ¡Debe cargar la web!
- Escribe branchserver.pt.pka -> ¡Debe cargar la web!

Ejercicio 2: Configuración de DHCPv6

Objetivo: Implementar direccionamiento dinámico IPv6 utilizando dos métodos distintos: Stateless (Sin estado) en el Router 1 y Stateful (Con estado) en el Router 2.

1. Configuración del Router R1 (DHCPv6 Stateless)

En este escenario, el router R1 entrega la configuración DNS mediante DHCP, pero deja que el cliente se autoconfigure su propia IP (SLAAC).

```
enable
configure terminal
ipv6 unicast-routing
```

```
! --- 1. Configuración de Interfaces y Enrutamiento ---
interface Serial0/1/0
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:ACAD:2::1/64
 clock rate 2000000
 no shutdown
 exit
```

```
! Ruta por defecto hacia R2
ipv6 route ::/0 2001:DB8:ACAD:2::2
```

```
! --- 2. Configuración del Pool DHCPv6 (Solo datos) ---
ipv6 dhcp pool R1-STATELESS
 dns-server 2001:DB8:ACAD::254
 domain-name Stateless.com
```

```
exit
```

```
! --- 3. Configuración de la Interfaz LAN ---
interface GigabitEthernet0/0/0
ipv6 address FE80::1 link-local
ipv6 address 2001:DB8:ACAD:1::1/64
! Bandera "O" (Other): Indica al PC que pida DNS al DHCP
ipv6 nd other-config-flag
! Vinculación del servidor DHCP
ipv6 dhcp server R1-STATELESS
no shutdown
exit
```

> Paso Crítico: Configuración del Cliente PC1

Para que la configuración surta efecto, el cliente debe solicitar la información.

1. Hacer clic en **PC1**.
2. Ir a la pestaña **Desktop > IP Configuration**.
3. En la sección **IPv6 Configuration**, seleccionar la opción **Automatic**.
 - Nota: Si ya estaba seleccionada, cambiar a **Static** y volver a **Automatic** para forzar una nueva solicitud.
4. **Resultado esperado:** El PC obtiene una IP autoconfigurada (SLAAC) y recibe el DNS y Dominio del servidor.

2. Configuración del Router R2 (DHCPv6 Stateful)

En este escenario, el router R2 actúa como un servidor DHCP completo, entregando tanto la dirección IP exacta como la configuración DNS.

```
enable
configure terminal
ipv6 unicast-routing
```

```
! --- 1. Configuración de Interfaces y Enrutamiento ---
interface Serial0/1/0
ipv6 address FE80::2 link-local
ipv6 address 2001:DB8:ACAD:2::2/64
no shutdown
exit
```

```
! Ruta por defecto hacia R1
ipv6 route ::/0 2001:DB8:ACAD:2::1
```

```
! --- 2. Configuración del Pool DHCPv6 (Completo) ---
ipv6 dhcp pool R2-STATEFUL
! Prefijo de direcciones a entregar
address prefix 2001:db8:acad:3:aaaa::/80
dns-server 2001:db8:acad::254
```

```
domain-name Stateful.com
```

```
exit
```

```
! --- 3. Configuración de la Interfaz LAN ---
```

```
interface GigabitEthernet0/0/0
```

```
 ipv6 address FE80::1 link-local
```

```
 ipv6 address 2001:DB8:ACAD:3::1/64
```

```
! Bandera "M" (Managed): Indica al PC que pida TODO al DHCP
```

```
 ipv6 nd managed-config-flag
```

```
! Vinculación del servidor DHCP
```

```
 ipv6 dhcp server R2-STATEFUL
```

```
no shutdown
```

```
exit
```

> Paso Crítico: Configuración del Cliente PC2

Es necesario forzar al cliente a negociar con el servidor DHCPv6 Stateful.

1. Hacer clic en **PC2**.
2. Ir a la pestaña **Desktop > IP Configuration**.
3. En la sección **IPv6 Configuration**, seleccionar la opción **Automatic**.
 - Nota: Al igual que antes, si ya estaba puesto, cambiar a **Static** y volver a **Automatic**.
4. **Resultado esperado:** El PC obtiene una IP específica del rango configurado (...:aaaa:...) y los datos de DNS/Dominio.

3. Verificación de Conectividad

Para confirmar el funcionamiento total de la red:

1. **Desde PC1:** Ping a la interfaz LAN de R2 -> ping 2001:db8:acad:3::1 (**Éxito**).
2. **Desde PC2:** Ping a la interfaz LAN de R1 -> ping 2001:db8:acad:1::1 (**Éxito**).

SLAAC y DHCPv6

| | SLAAC | SLAAC + DHCPv6 sin estado | DHCPv6 con estado |
|---|--------------|----------------------------------|--------------------------|
| A | 1 | 1 | 0 |
| O | 0 | 1 | 0 |
| M | 0 | 0 | 1 |

PRÁCTICA 8

Ejercicio 1

Abre el fichero parte1.pcap. Observa los tipos de paquetes que se tiene, centra la atención en el primer paquete recibido y consultando la cabecera relativa al protocolo IP, rellena los siguientes datos del mismo:

| Primer paquete IP recibido | |
|----------------------------|-----------------|
| Dirección IP origen | 150.214.117.76 |
| Dirección IP destino | 150.214.117.255 |
| Protocolo | UDP (17) |
| Tamaño cabecera | 20 bytes |
| Tamaño total | 78 |
| TTL | 128 |
| Identificador | 0x37bf (14271) |

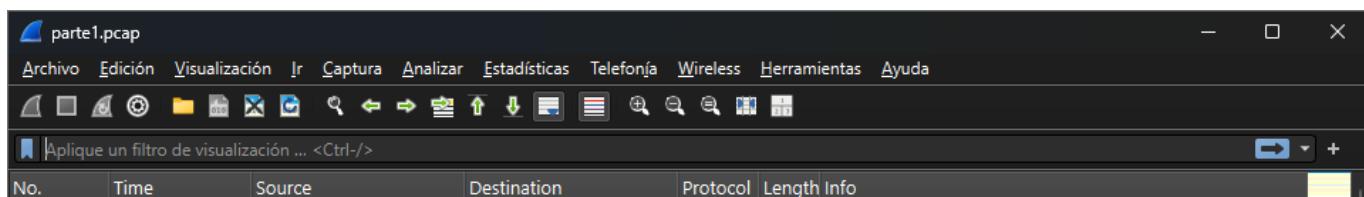
```
▼ Internet Protocol Version 4, Src: 150.214.117.76, Dst: 150.214.117.255
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ▼ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    0000 00.. = Differentiated Services Codepoint: Default (0)
    .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 78
    Identification: 0x37bf (14271)
  ▼ 000. .... = Flags: 0x0
    0.... .... = Reserved bit: Not set
    .0... .... = Don't fragment: Not set
    ..0. .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 128
    Protocol: UDP (17)
    Header Checksum: 0xe9e7 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 150.214.117.76
    Destination Address: 150.214.117.255
    [Stream index: 0]
```

Nota: Nosotros no podemos capturar y esto son filtros para la creación del fichero captura los de abajo son el filtro que solo nos aparezcan lo buscado

| Ejercicio | Enunciado / Requisito | Respuesta (Sintaxis del Filtro) | Explicación Técnica |
|-----------|-----------------------|---------------------------------|---------------------|
| | | | |

| | | | |
|-------------|---|--|---|
| Ejercicio 2 | Aceptar solo paquetes que tengan como origen o destino tu dirección IP. | host 150.214.117.76 <i>ip.addr == IP_NUESTRA_MÁQUINA</i> | La primitiva host filtra bidireccionalmente (tanto si la IP es origen como si es destino). (<i>Sustituir la IP por la de tu máquina</i>). |
| Ejercicio 3 | Capturar solo consultas DNS (origen o destino puerto 53 UDP). | udp port 53 <i>udp.port == 53</i> | Filtrar el protocolo de transporte UDP y el puerto específico del servicio DNS. |
| Ejercicio 4 | Capturar consultas DNS (UDP 53) cuyo origen sea tu máquina. | udp port 53 and src host 150.214.117.76 <i>udp.port == 53 and ip.src == 150.214.117.76</i> | Se combinan dos primitivas con el operador lógico <i>and</i> <i>src</i> fuerza a que la IP sea solo de origen. |
| Ejercicio 5 | Capturar solamente paquetes UDP. | udp <i>udp</i> | Filtrar por protocolo de nivel de transporte, descartando TCP, ICMP, etc. |

CAPTURA DE LOS EJERCICIOS



Es ahí dónde se hace la búsqueda de los filtros.

Protocolo ICMP

Ejercicio 1

Abre el archivo parte1.pcap, en la captura mostrada se encuentra el envío producido por el comando ping -c 4 209.85.229.104.

Para facilitar el análisis, es recomendable aplicar un filtro de presentación para que sólo aparezcan los paquetes enviados a la dirección 209.85.229.104. Para ello, bastará con escribir (ip.addr == 209.85.229.104) en el campo de texto de filtro de la ventana principal y pulsar el botón Apply.

¿Cuántos mensajes ICMP se producen, prestando especial atención a los campos tipo, código, y bytes de datos? Asimismo, analiza las cabeceras IP de cada uno de ellos, y en concreto los campos longitud de la cabecera y longitud total

1. ¿Cuántos mensajes ICMP se producen? Observando el listado de paquetes en tu captura, se ven 4 intercambios completos de ping (solicitud y respuesta).

- **Total de mensajes: 8 mensajes (4 Echo request y 4 Echo reply).**
- Se corresponden con los paquetes: 121-122, 131-134, 144-145 y 156-157.

2. Análisis de los campos ICMP (Tipo, Código y Bytes de datos) Mirando el detalle del **Paquete 121** (que es una solicitud request) que tienes seleccionado:

- **Tipo (Type):** 8 (Echo (ping) request). Nota: En las respuestas (reply) este valor será 0.
- **Código (Code):** 0.
- **Bytes de datos:** 48 bytes (Se ve abajo del todo en "Data (48 bytes)").

3. Análisis de la cabecera IP Mirando la sección "Internet Protocol Version 4" del mismo paquete:

- **Longitud de la cabecera (Header Length):** 20 bytes.
- **Longitud total (Total Length):** 84.

The screenshot shows the Wireshark interface with the following details:

- File:** parte1.pcap
- Filter:** ip.addr == 209.85.229.104
- Packets:** 157 (total), 121 (selected)
- Selected Packet (Packet 121):**
 - Ethernet II:** Src: Apple_14:9d:56 (c4:2c:03:14:9d:56), Dst: All-HSRP-routers_12 (00:00:0c:07:ac:12)
 - Destination: All-HSRP-routers_12 (00:00:0c:07:ac:12)
 - 0. = LG bit: Globally unique address (factory default)
 - 0. = IG bit: Individual address (unicast)
 - Source:** Apple_14:9d:56 (c4:2c:03:14:9d:56)
 - 0. = LG bit: Globally unique address (factory default)
 - 0. = IG bit: Individual address (unicast)Type: IPv4 (0x0800)
[Stream index: 31]
 - Internet Protocol Version 4:** Src: 192.168.117.203, Dst: 209.85.229.104
 - 0100 = Version: 4
 - 0101 = Header Length: 20 bytes (5)
 - Differentiated Services Field: 0x00 (DSSCP: CS0, ECN: Not-ECT)
 - 0000 00.. = Differentiated Services Codepoint: Default (0)
 - 0000 = Explicit Congestion Notification: Not ECN-Capable Transport (0)Total Length: 84
 - Identification: 0x89cf (35279)
 - 000. = Flags: 0x0
 - 0.... = Reserved bit: Not set
 - .0.... = Don't fragment: Not set
 - .0.... = More fragments: Not set
 - ... 0 0000 0000 0000 = Fragment Offset: 0
 - Time to Live: 64
 - Protocol: ICMP (1)
 - Header Checksum: 0x0000 [validation disabled]
[Header checksum status: Unverified]
 - Source Address: 192.168.117.203
 - Destination Address: 209.85.229.104
[Stream index: 27]
 - Internet Control Message Protocol**
 - Type: Echo (ping) request (8)
 - Code: 0

Ejercicio 2

Para comprobar el funcionamiento del protocolo en otras condiciones, en el mismo fichero que tenemos abierto se encuentra la captura relativa a la ejecución del comando ping -l (tamaño-paquete) –c (número-de-paquetes) 192.168.117.205. Para controlar el número de mensajes que se envían sería conveniente que en el filtro de presentación se indicase (ip.addr == 192.168.117.205).

¿Cuántos mensajes ICMP se producen ahora en cada envío y recepción?, ¿cuántos paquetes se han enviado? Analiza los parámetros de la cabecera que te indican cómo están fragmentados los mensajes, comprobar para ello los campos don't fragment, more fragment y fragment offset de un envío y de una recepción.

1. ¿Cuántos mensajes ICMP (paquetes IP) se producen ahora en cada envío y recepción?

Observando la lista de paquetes en tu imagen (por ejemplo, los paquetes 418, 419 y 420), vemos que un solo "Ping Request" se divide en 3 fragmentos.

- **Envío (Request):** 3 paquetes IP.
- **Recepción (Reply):** 3 paquetes IP.

2. ¿Cuántos paquetes se han enviado?

En la captura visible se distinguen 3 secuencias completas de Ping (identificadas por los IDs 0xaf93, 0xb549 y 0x0e18).

- Como cada ping consta de 3 fragmentos de ida y 3 de vuelta, en total para esa secuencia visible se han movido $3 \times 3 = 9$ paquetes de envío.

3. Análisis de los parámetros de fragmentación (Cabeza IP)

Analizando los campos que se ven en tu imagen (específicamente el paquete 420, que es el último fragmento de una petición):

- **Flags - Don't fragment (No fragmentar):** Está en **Not set (0)**.
 - *Explicación:* Esto es obligatorio. Si estuviera activo (1), el router descartaría el paquete porque es demasiado grande y no le permiten partirllo.
- **Flags - More fragments (Más fragmentos):**
 - En los dos primeros paquetes del grupo (ej. 418 y 419): Este campo vale **1 (Set)**, indicando que "viene más trozos detrás".
 - En el último paquete (el 420 que tienes seleccionado): Este campo vale **0 (Not set)**, indicando que "este es el final del mensaje".
- **Fragment Offset (Desplazamiento del fragmento):** Indica en qué posición va ese trozo de datos.
 - Primer paquete (418): Offset **0**.
 - Segundo paquete (419): Offset **1480**.
 - Tercer paquete (420): Offset **2960** (como se ve en tu panel de detalles).

parte1.pcap

Archivo Edición Visualización Ir Captura Analizar Estadísticas Telefonía Wireless Herramientas Ayuda

ip.addr == 192.168.117.205

| No. | Time | Source | Destination | Protocol | Length | Info |
|-------|-----------|-----------------|-----------------|----------|--------|--|
| 73 | 6.908176 | 192.168.117.205 | 255.255.255.255 | DB-LSP.. | 151 | Dropbox LAN sync Discovery Protocol, JSON |
| 74 | 6.908331 | 192.168.117.205 | 192.168.117.205 | DB-LSP.. | 151 | Dropbox LAN sync Discovery Protocol, JSON |
| 361 | 35.680510 | 192.168.117.203 | 192.168.117.205 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=0, ID=a9f93) [Reassembled in #363] |
| 362 | 35.680670 | 192.168.117.203 | 192.168.117.205 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=1480, ID=a9f93) [Reassembled in #363] |
| 363 | 35.680677 | 192.168.117.203 | 192.168.117.205 | ICMP | 154 | Echo (ping) request id=0x3a02, seq=1/256, ttl=64 (reply in 366) |
| 364 | 35.681527 | 192.168.117.203 | 192.168.117.205 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=0, ID=5edb) [Reassembled in #366] |
| 365 | 35.681718 | 192.168.117.203 | 192.168.117.205 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=1480, ID=5edb) [Reassembled in #366] |
| 366 | 35.681723 | 192.168.117.203 | 192.168.117.205 | ICMP | 154 | Echo (ping) reply id=0x3a02, seq=1/256, ttl=64 (request in 363) |
| 379 | 36.961722 | 192.168.117.205 | 255.255.255.255 | DB-LSP.. | 151 | Dropbox LAN sync Discovery Protocol, JSON |
| 380 | 36.961756 | 192.168.117.205 | 192.168.117.205 | DB-LSP.. | 151 | Dropbox LAN sync Discovery Protocol, JSON |
| * 418 | 41.009722 | 192.168.117.203 | 192.168.117.205 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=0, ID=b549) [Reassembled in #420] |
| * 419 | 41.009885 | 192.168.117.203 | 192.168.117.205 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=1480, ID=b549) [Reassembled in #420] |
| + 420 | 41.009895 | 192.168.117.203 | 192.168.117.205 | ICMP | 154 | Echo (ping) request id=0xb02, seq=0/0, ttl=64 (reply in 423) |
| 421 | 41.018792 | 192.168.117.205 | 192.168.117.203 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=0, ID=5edc) [Reassembled in #423] |
| 422 | 41.018870 | 192.168.117.205 | 192.168.117.203 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=1480, ID=5edc) [Reassembled in #423] |
| + 423 | 41.018876 | 192.168.117.205 | 192.168.117.203 | ICMP | 154 | Echo (ping) reply id=0xb02, seq=0/0, ttl=64 (request in 420) |
| 432 | 42.010069 | 192.168.117.203 | 192.168.117.205 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=0, ID=0e18) [Reassembled in #434] |
| 433 | 42.010233 | 192.168.117.203 | 192.168.117.205 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=1480, ID=0e18) [Reassembled in #434] |
| 434 | 42.010244 | 192.168.117.203 | 192.168.117.205 | ICMP | 154 | Echo (ping) request id=0xb02, seq=1/256, ttl=64 (reply in 437) |
| 435 | 42.011169 | 192.168.117.205 | 192.168.117.203 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=0, ID=5edd) [Reassembled in #437] |
| 436 | 42.011209 | 192.168.117.205 | 192.168.117.203 | IPv4 | 1514 | Fragmented IP protocol (proto=ICMP 1, offf=1480, ID=5edd) [Reassembled in #437] |
| 437 | 42.011215 | 192.168.117.205 | 192.168.117.203 | ICMP | 154 | Echo (ping) reply id=0xb02, seq=1/256, ttl=64 (request in 434) |

```

000. .... = Flags: 0x0
  0... .... = Reserved bit: Not set
  .0... .... = Don't fragment: Not set
  ..0. .... = More fragments: Not set
  ...0 0001 0111 0010 = Fragment Offset: 2960
  ...

```

Protocolo DNS

El protocolo DNS, que emplea el puerto 53 de UDP, se emplea para poder denominar a los computadores mediante nombres simbólicos, sin tener que recordar las direcciones IP correspondientes a cada computador. A continuación, observaremos el funcionamiento del DNS.

Ejercicio 3

Continuando con el fichero parte1.pcap, vamos a analizar la información que se genera en la captura cuando se ejecuta el comando ping -c 1 www.google.es.

Para facilitar el análisis, es recomendable aplicar un filtro de presentación para que sólo aparezcan los paquetes de este protocolo (dns).

¿Cuántos mensajes DNS se generan? ¿Qué tipo de consulta se realiza? El tipo de las consultas viene detallado en el campo Queries de la cabecera del protocolo DNS. ¿En qué puerto y dirección IP está localizado el servidor de nombres? ¿Qué dirección IP tiene el dominio www.google.es? ¿Qué protocolo de capa de transporte utiliza los paquetes DNS?

1. ¿Cuántos mensajes DNS se generan? En tu lista de paquetes (parte superior de la imagen) aparecen 2 mensajes:

- El paquete 119 (Solicitud / Query).
- El paquete 120 (Respuesta / Response).

2. ¿Qué tipo de consulta se realiza? Mirando el panel de detalles del paquete 119 (abajo), en la sección **Domain Name System (query) -> Queries:**

- El tipo es **Type A** (Host Address). Esto significa que está preguntando por la dirección IPv4 del dominio.

3. ¿En qué puerto y dirección IP está localizado el servidor de nombres? Mirando las cabeceras del paquete de solicitud (119):

- **Dirección IP:** 150.214.110.3 (Es la dirección *Destination* en la solicitud y *Source* en la respuesta).
- **Puerto:** 53 (Se ve en la línea "User Datagram Protocol" -> *Dst Port*: 53).

4. ¿Qué dirección IP tiene el dominio www.google.es? Aunque tienes seleccionado el paquete de pregunta, la respuesta se puede leer en la columna "Info" del **paquete 120** (el renglón azul justo debajo):

- Ahí dice: Standard query response ... A 209.85.229.104
- Por tanto, la dirección IP es **209.85.229.104**.

5. ¿Qué protocolo de capa de transporte utiliza los paquetes DNS? Se ve claramente en el panel central de detalles:

- **UDP** (User Datagram Protocol).

| No. | Time | Source | Destination | Protocol |
|-----|-----------|-----------------|-----------------|----------|
| 119 | 11.070543 | 192.168.117.203 | 150.214.110.3 | DNS |
| 120 | 11.071167 | 150.214.110.3 | 192.168.117.203 | DNS |

Protocolo ARP

Para que un datagrama llegue a su destino es necesario especificar, además de la dirección IP destino, la dirección física del adaptador de red que debe recibir la trama en la que viaja el datagrama. Este adaptador de red puede ser el del host destino o bien el de un router intermedio. Precisamente, para averiguar la dirección física que corresponde a una dirección IP determinada se creó el protocolo ARP, como se vio en la práctica anterior.

Ejercicio 4

En el fichero parte1.pcap, se encuentran almacenados los paquetes producidos por la ejecución del comando **ping -c 3 192.168.117.205**. Para facilitar el análisis, es recomendable aplicar un filtro de presentación para que sólo aparezcan los paquetes del protocolo ARP (arp).

1. ¿Se han generado paquetes ARP para resolver la dirección IP indicada?

Sí.

- El paquete 352 es la solicitud ("Who has 192.168.117.205?").
- El paquete 353 es la respuesta ("192.168.117.205 is at...").

2. Análisis del mensaje de petición (Paquete 352) *Aunque no tienes desplegado el detalle del 352 en las fotos, podemos sacar los datos de la lista y cruzarlos con la información del paquete 416 que sí muestras, ya que es la misma máquina.*

- **¿Cuál es la dirección MAC del remitente (Sender MAC)?** Es la dirección física de tu máquina (la que hace el ping).
 - Respuesta: c4:2c:03:14:9d:56 (Corresponde al fabricante Apple).

3. Determinación del tipo de mensaje

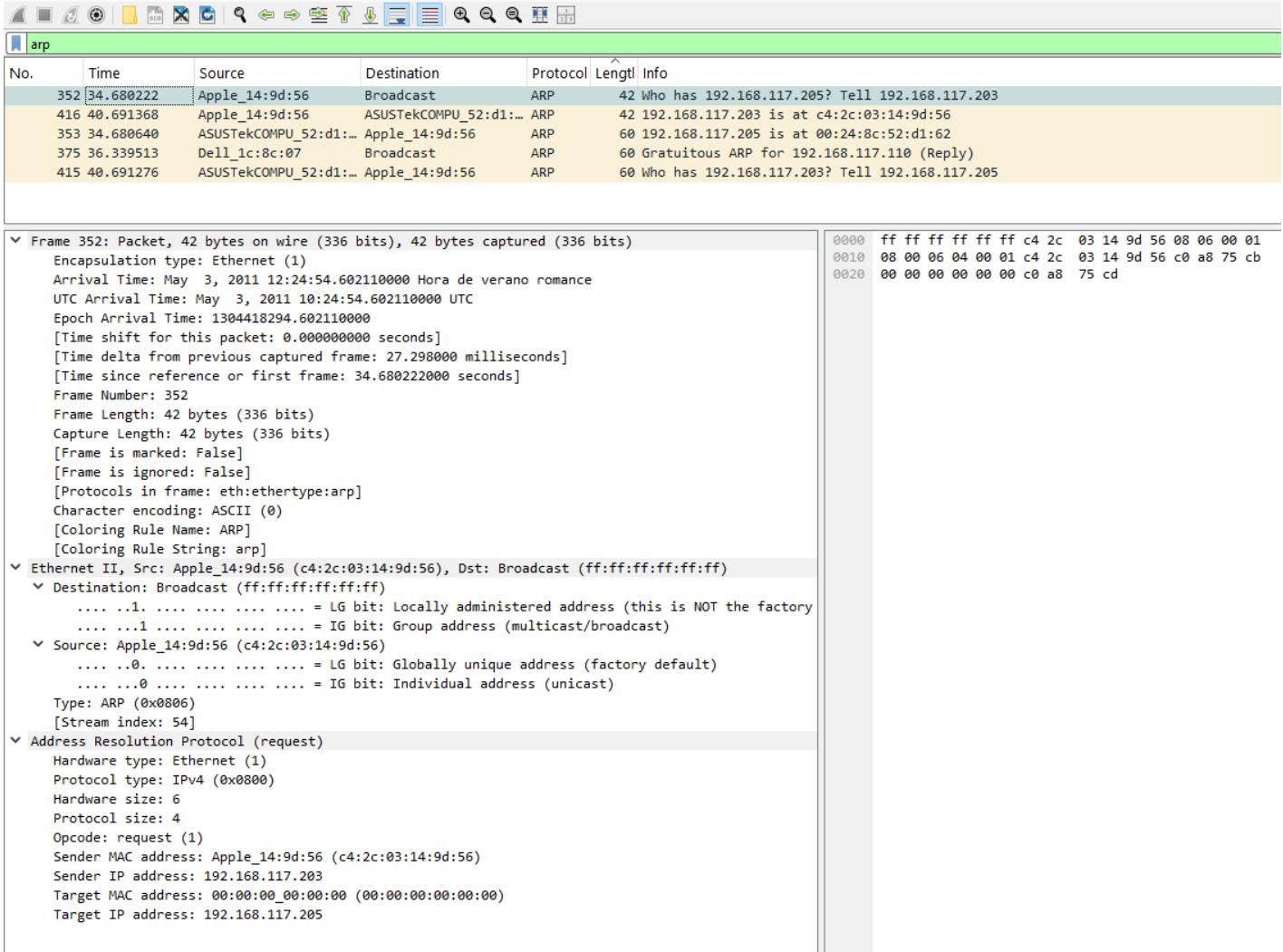
- ¿Qué campo nos da esta información? El campo se llama Opcode (Código de operación).
- Valor: Si miras el detalle de cualquier solicitud (como el paquete 415 de tu segunda foto), verás que pone Opcode: request (1).

4. Análisis del mensaje de respuesta (Paquete 353) *Para esta respuesta usamos la imagen image_f8f05c.png donde tienes seleccionado el paquete 353.*

- Dirección física (MAC) del destinatario: Mirando el campo Target MAC address:
 - Respuesta: c4:2c:03:14:9d:56 (Apple).
- Dirección IP del destinatario: Mirando el campo Target IP address:
 - Respuesta: 192.168.117.203.
- ¿A quién corresponde dicha dirección física? Corresponde a tu equipo (el que inició la comunicación y lanzó el ping), que ahora recibe la respuesta con la dirección MAC que necesitaba para enviar los datos.

parte1.pcap

Archivo Edición Visualización Ir Captura Analizar Estadísticas Telefonía Wireless Herramientas Ayuda



No. Time Source Destination Protocol Length Info

352 34.680222 Apple_14:9d:56 Broadcast ARP 42 Who has 192.168.117.205? Tell 192.168.117.203
 416 40.691368 Apple_14:9d:56 ASUSTekCOMPU_52:d1... ARP 42 192.168.117.203 is at c4:2c:03:14:9d:56
 353 34.680640 ASUSTekCOMPU_52:d1... Apple_14:9d:56 ARP 60 192.168.117.205 is at 00:24:8c:52:d1:62
 375 36.339513 Dell_1c:8c:07 Broadcast ARP 60 Gratuitous ARP for 192.168.117.110 (Reply)
 415 40.691276 ASUSTekCOMPU_52:d1... Apple_14:9d:56 ARP 60 Who has 192.168.117.203? Tell 192.168.117.205

Frame 352: Packet, 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
 Encapsulation type: Ethernet (1)
 Arrival Time: May 3, 2011 12:24:54.602110000 Hora de verano romance
 UTC Arrival Time: May 3, 2011 10:24:54.602110000 UTC
 Epoch Arrival Time: 1304418294.602110000
 [Time shift for this packet: 0.000000000 seconds]
 [Time delta from previous captured frame: 27.298000 milliseconds]
 [Time since reference or first frame: 34.680222000 seconds]
 Frame Number: 352
 Frame Length: 42 bytes (336 bits)
 Capture Length: 42 bytes (336 bits)
 [Frame is marked: False]
 [Frame is ignored: False]
 [Protocols in frame: eth:ethertype:arp]
 Character encoding: ASCII (0)
 [Coloring Rule Name: ARP]
 [Coloring Rule String: arp]

Ethernet II, Src: Apple_14:9d:56 (c4:2c:03:14:9d:56), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
 Destination: Broadcast (ff:ff:ff:ff:ff:ff)
1. = LG bit: Locally administered address (this is NOT the factory
1. = IG bit: Group address (multicast/broadcast)
 Source: Apple_14:9d:56 (c4:2c:03:14:9d:56)
0. = LG bit: Globally unique address (factory default)
0. = IG bit: Individual address (unicast)
 Type: ARP (0x0806)
 [Stream index: 54]

Address Resolution Protocol (request)
 Hardware type: Ethernet (1)
 Protocol type: IPv4 (0x0800)
 Hardware size: 6
 Protocol size: 4
 Opcode: request (1)
 Sender MAC address: Apple_14:9d:56 (c4:2c:03:14:9d:56)
 Sender IP address: 192.168.117.203
 Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
 Target IP address: 192.168.117.205

| | | |
|------|-------------------------|-------------------------|
| 0000 | ff ff ff ff ff c4 2c | 03 14 9d 56 08 06 00 01 |
| 0010 | 08 00 06 04 00 01 c4 2c | 03 14 9d 56 c0 a8 75 cb |
| 0020 | 00 00 00 00 00 00 c0 a8 | 75 cd |

Protocolo UDP

Es el protocolo no confiable empleado en la capa de transporte de TCP/IP.

Ejercicio 5

Para analizar la estructura de este protocolo con Wireshark vamos a abrir el archivo udp.pcap donde se encuentra almacenado la captura de una comunicación UDP.

Analice los paquetes y comente cuáles son las direcciones IP y puertos involucrados en la comunicación.

¿Cuál es el número de paquetes UDP y el número de bytes de datos intercambiados?, ¿qué datos se han mandado?

1. Direcciones IP y puertos involucrados

Mirando la cabecera IP y la cabecera UDP de cualquiera de los dos paquetes (por ejemplo, el paquete 3 en image_eeff1c.png):

- Dirección IP Origen: 10.0.0.2
- Dirección IP Destino: 11.0.0.2
- Puerto Origen: 32768
- Puerto Destino: 33000

2. Número de paquetes, bytes y datos enviados

Observando la lista de paquetes en la parte superior y el campo "Data" en la parte inferior de ambas capturas:

- **Número de paquetes UDP:** Son 2 paquetes (El paquete nº 3 y el paquete nº 4 de la lista). Los dos primeros son ARP.
- **Datos mandados:**
 - En el primer paquete UDP (Paquete 3), el contenido de los datos es "hola" (seguido de un salto de línea). Se ve en la sección de datos hexadecimales a la derecha: 68 6f 6c 61 0a.
 - En el segundo paquete UDP (Paquete 4), el contenido es "adios" (seguido de un salto de línea).
- **Número de bytes de datos intercambiados:**
 - El primero lleva 5 bytes ("hola" + enter).
 - El segundo lleva 6 bytes ("adios" + enter).
 - Total: 11 bytes de datos.

The screenshot shows the Wireshark interface with the file 'udp.cap' loaded. The main window displays a list of network frames. Frame 3 is selected, showing details for an ARP request and a UDP packet. The UDP packet has a source of 10.0.0.2 and a destination of 11.0.0.2, with a length of 47 bytes. The data payload is shown as 'hola' in ASCII and its hex representation (68 6f 6c 61 0a) in the hex dump column. Frame 4 follows, showing an ARP response and another UDP packet with the same source and destination, containing the data 'adios' (68 6f 6c 61 0a 0d 0a) in its hex dump.

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|-------------------|-------------------|----------|--------|----------------------------------|
| 1 | 0.000000 | 8e:d5:68:dd:a2:ce | Broadcast | ARP | 42 | Who has 10.0.0.1? Tell 10.0.0.2 |
| 2 | 0.000105 | de:02:94:29:fc:d4 | 8e:d5:68:dd:a2:ce | ARP | 42 | 10.0.0.1 is at de:02:94:29:fc:d4 |
| 3 | 0.000188 | 10.0.0.2 | 11.0.0.2 | UDP | 47 | 32768 → 33000 Len=5 |

Frame 3: Packet, 47 bytes on wire (376 bits), 47 bytes captured (376 bits)
Encapsulation type: Ethernet (1)
Arrival Time: Feb 23, 2008 10:20:58.527236000 Hora estándar romance
UTC Arrival Time: Feb 23, 2008 09:20:58.527236000 UTC
Epoch Arrival Time: 1203758458.527236000
[Time shift for this packet: 0.000000000 seconds]
[Time delta from previous captured frame: 83.000 microseconds]
[Time delta from previous displayed frame: 83.000 microseconds]
[Time since reference or first frame: 188.000 microseconds]
Frame Number: 3
Frame Length: 47 bytes (376 bits)
Capture Length: 47 bytes (376 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: eth:ethertype:ip:udp:data]
Character encoding: ASCII (0)
[Coloring Rule Name: UDP]
[Coloring Rule String: udp]
Ethernet II, Src: 8e:d5:68:dd:a2:ce (8e:d5:68:dd:a2:ce), Dst: de:02:94:29:fc:d4 (de:02:94:29:fc:d4)
Destination: de:02:94:29:fc:d4 (de:02:94:29:fc:d4)
....1. = LG bit: Locally administered address (this is NOT the factory
....0. = IG bit: Individual address (unicast)
Source: 8e:d5:68:dd:a2:ce (8e:d5:68:dd:a2:ce)
....1. = LG bit: Locally administered address (this is NOT the factory
....0. = IG bit: Individual address (unicast)
Type: IPv4 (0x0800)
[Stream index: 1]
Internet Protocol Version 4, Src: 10.0.0.2, Dst: 11.0.0.2
0100 = Version: 4
....0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
0000 00.. = Differentiated Services Codepoint: Default (0)
.... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
Total Length: 33
Identification: 0x05d (26717)
010. = Flags: 0x2, Don't fragment
0.... = Reserved bit: Not set
.1. = Don't fragment: Set
.0. = More fragments: Not set
..0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 64
Protocol: UDP (17)
Header Checksum: 0xbdb6 [validation disabled]
[Header checksum status: Unverified]
Source Address: 10.0.0.2
Destination Address: 11.0.0.2
[Stream index: 0]
User Datagram Protocol, Src Port: 32768, Dst Port: 33000
Source Port: 32768
Destination Port: 33000
Length: 13
Checksum: 0xb17 [unverified]
[Checksum Status: Unverified]
[Stream index: 0]
[Stream Packet Number: 1]
[Timestamps]
[Time since first frame: 0.000000000 seconds]
[Time since previous frame: 0.000000000 seconds]
UDP payload (5 bytes)
Data (5 bytes)
Data: 686f6c610a
[Length: 5]
Data (6 bytes)
Data: 6164696f730a
[Length: 6]

Protocolo TCP

Es el protocolo confiable empleado en la capa de transporte de TCP/IP.

Ejercicio 6

Para analizar la estructura de este protocolo con Wireshark vamos a abrir el archivo tcp.pcap donde se encuentra almacenado la captura de una comunicación TCP.

Analice los paquetes y responda, suponiendo que se emplea la comunicación cliente/servidor ¿Cuál es la dirección IP y el puerto del cliente TCP?, y ¿la dirección IP y el puerto del servidor TCP?

¿Cuántos segmentos TCP se han enviado desde el cliente al servidor? y ¿desde el servidor al cliente?

¿Qué extremo cierra primero la conexión?

En el menú de Wireshark, seleccionando Edit → Preferences → Protocols → TCP, puedes desactivar la opción Relative Sequence Numbers. De esta forma podrás observar los números de secuencia reales, en lugar de los números relativos que muestra por omisión Wireshark.

¿Cuántos bytes de datos envía el cliente al servidor? Indica cuáles son los números de secuencia del SYN y del FIN que envía el cliente y qué relación tienen con la cantidad de datos enviada al servidor.

¿Cuántos bytes de datos envía el servidor al cliente? Indica cuáles son los números de secuencia del SYN y del FIN que envía el servidor y qué relación tienen con la cantidad de datos enviada al servidor.

Cuando hayas observado los números de secuencia reales, vuelve a activar la opción Relative Sequence Numbers y vuelve a contestar las últimas dos preguntas realizadas

1. Identificación de Cliente y Servidor

Mirando el primer paquete (paquete nº 3), que lleva la etiqueta [SYN]:

- Cliente TCP: Es quien inicia la conexión.
 - Dirección IP: 10.0.0.2
 - Puerto: 60709
- Servidor TCP: Es quien recibe la petición.
 - Dirección IP: 11.0.0.2
 - Puerto: 34000

2. Recuento de segmentos

Contando los paquetes en el listado de tu imagen (del 3 al 12):

- Del Cliente al Servidor (10.0.0.2 → 11.0.0.2): Son los paquetes 3, 5, 6, 8, 10, 12.
 - Total: 6 segmentos.
- Del Servidor al Cliente (11.0.0.2 → 10.0.0.2): Son los paquetes 4, 7, 9, 11.
 - Total: 4 segmentos.

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|-------------------|-------------------|----------|--------|---|
| 1 | 0.000000 | 8e:d5:68:dd:a2:ce | Broadcast | ARP | 42 | Who has 10.0.0.1? Tell 10.0.0.2 |
| 2 | 0.000548 | de:02:94:29:fc:d4 | 8e:d5:68:dd:a2:ce | ARP | 42 | 10.0.0.1 is at de:02:94:29:fc:d4 |
| 3 | 0.000134 | 10.0.0.2 | 11.0.0.2 | TCP | 74 | 60709 → 34000 [SYN] Seq=1367801698 Win=5840 Len=0 MSS=1460 TStamp=125335 TSectr=0 WS=2 |
| 4 | 0.000918 | 11.0.0.2 | 10.0.0.2 | TCP | 74 | 34000 → 60709 [SYN, ACK] Seq=1363273423 Ack=1367801699 Win=5792 Len=0 MSS=1460 TStamp=125319 TSectr=125335 WS=2 |
| 5 | 0.010351 | 10.0.0.2 | 11.0.0.2 | TCP | 66 | 60709 → 34000 [ACK] Seq=1367801699 Ack=1363273424 Win=5840 Len=0 TStamp=125341 TSectr=125319 |
| 6 | 1.962313 | 10.0.0.2 | 11.0.0.2 | TCP | 71 | 60709 → 34000 [PSH, ACK] Seq=1367801699 Ack=1363273424 Win=5840 Len=5 TStamp=125537 TSectr=125319 |
| 7 | 1.963330 | 11.0.0.2 | 10.0.0.2 | TCP | 66 | 34000 → 60709 [ACK] Seq=1363273424 Ack=1367801704 Win=5792 Len=0 TStamp=125516 TSectr=125537 |
| 8 | 3.251756 | 10.0.0.2 | 11.0.0.2 | TCP | 72 | 60709 → 34000 [PSH, ACK] Seq=1367801704 Ack=1363273424 Win=5840 Len=6 TStamp=125666 TSectr=125516 |
| 9 | 3.252305 | 11.0.0.2 | 10.0.0.2 | TCP | 66 | 34000 → 60709 [ACK] Seq=1363273424 Ack=1367801710 Win=5792 Len=0 TStamp=125645 TSectr=125666 |
| 10 | 4.170910 | 10.0.0.2 | 11.0.0.2 | TCP | 66 | 60709 → 34000 [FIN, ACK] Seq=1367801710 Ack=1363273424 Win=5840 Len=0 TStamp=125758 TSectr=125645 |
| 11 | 4.172336 | 11.0.0.2 | 10.0.0.2 | TCP | 66 | 34000 → 60709 [FIN, ACK] Seq=1363273424 Ack=1367801711 Win=5792 Len=0 TStamp=125758 TSectr=125737 |
| 12 | 4.172611 | 10.0.0.2 | 11.0.0.2 | TCP | 71 | 60709 → 34000 [ACK] Seq=1367801711 Ack=1363273425 Win=5840 Len=0 TStamp=125758 TSectr=125737 |
| 13 | 4.989294 | de:02:94:29:fc:d4 | 8e:d5:68:dd:a2:ce | ARP | 42 | Who has 10.0.0.2? Tell 10.0.0.1 |
| 14 | 4.990194 | 8e:d5:68:dd:a2:ce | de:02:94:29:fc:d4 | ARP | 42 | 10.0.0.2 is at 8e:d5:68:dd:a2:ce |

3. ¿Qué extremo cierra primero la conexión?

El cierre se indica con el flag FIN.

- Mirando la columna "Info", el paquete 10 es el primero que muestra [FIN, ACK].
- El origen de este paquete es 10.0.0.2.
- Respuesta: El Cliente cierra primero.

Análisis de Números de Secuencia (Reales / Absolutos)

Como en tu imagen ya se ven los números largos, respondemos primero a la parte del ejercicio que pide desactivar "Relative Sequence Numbers".

4. Datos enviados por el Cliente (10.0.0.2)

- Bytes de datos:** Si sumamos la longitud (Len) de los paquetes de datos del cliente (Paq. 6 Len=5 y Paq. 8 Len=6), el total es 11 bytes ("hola" + "adios").
- Secuencia del SYN (Paquete 3):** 1367801698.
- Secuencia del FIN (Paquete 10):** Aunque no se ve el detalle, se calcula sumando al SYN el 1 inicial + los 11 bytes de datos.
 - Valor calculado: 1367801710.
- Relación:** La diferencia entre el FIN y el SYN es 12 ($1367801710 - 1367801698 = 12$).
 - Esto corresponde a 11 bytes de datos + 1 (el flag SYN consume un número).

5. Datos enviados por el Servidor (11.0.0.2)

- Bytes de datos: Todos sus paquetes (4, 7, 9, 11) tienen Len=0. Por tanto, envía 0 bytes de datos.
- Secuencia del SYN (Paquete 4): 1363273423 (visible en la columna Info).
- Secuencia del FIN (Paquete 11): Al no haber datos, solo aumenta 1 por el SYN inicial.
 - Valor calculado: 1363273424.
- Relación: La diferencia es 1 (0 bytes de datos + 1 por el SYN).

Análisis con Números Relativos (Si activamos la opción)

El ejercicio pide contestar esto de nuevo suponiendo que activas "Relative Sequence Numbers" (donde Wireshark pone el inicio a 0 para facilitar la lectura).

- Cliente:
 - Seq SYN: 0
 - Seq FIN: 12
- Servidor:
 - Seq SYN: 0
 - Seq FIN: 1

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|-------------------|-------------------|----------|--------|---|
| 1 | 0.000000 | 8e:d5:68:dd:a2:ce | Broadcast | ARP | 42 | Who has 10.0.0.1? Tell 10.0.0.2 |
| 2 | 0.000548 | de:02:94:29:fc:d4 | 8e:d5:68:dd:a2:ce | ARP | 42 | 10.0.0.1 is at de:02:94:29:fc:d4 |
| 3] | 0.000134 | 10.0.0.2 | 11.0.0.2 | TCP | 74 | 60709 → 34000 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TStamp=125335 TSectr=0 WS=2 |
| 4 | 0.000918 | 11.0.0.2 | 10.0.0.2 | TCP | 74 | 34000 → 60709 [SYN, ACK] Seq=6 Ack=1 Win=5792 Len=0 MSS=1460 TStamp=125335 TSectr=125335 WS=2 |
| 5 | 0.010351 | 10.0.0.2 | 11.0.0.2 | TCP | 66 | 60709 → 34000 [ACK] Seq=1 Ack=1 Win=5840 Len=0 TStamp=125341 TSectr=125319 |
| 6 | 1.962313 | 10.0.0.2 | 11.0.0.2 | TCP | 71 | 60709 → 34000 [PSH, ACK] Seq=1 Ack=1 Win=5840 Len=5 TStamp=125337 TSectr=125319 |
| 7 | 1.963330 | 11.0.0.2 | 10.0.0.2 | TCP | 66 | 34000 → 60709 [ACK] Seq=1 Ack=6 Win=5792 Len=0 TStamp=125516 TSectr=125537 |
| 8 | 3.251756 | 10.0.0.2 | 11.0.0.2 | TCP | 72 | 60709 → 34000 [PSH, ACK] Seq=6 Ack=1 Win=5844 Len=6 TStamp=125666 TSectr=125516 |
| 9 | 3.252305 | 11.0.0.2 | 10.0.0.2 | TCP | 66 | 34000 → 60709 [ACK] Seq=1 Ack=12 Win=5792 Len=0 TStamp=125645 TSectr=125666 |
| 10 | 4.170910 | 10.0.0.2 | 11.0.0.2 | TCP | 66 | 60709 → 34000 [FIN, ACK] Seq=12 Ack=1 Win=5840 Len=0 TStamp=125758 TSectr=125645 |
| 11 | 4.172336 | 11.0.0.2 | 10.0.0.2 | TCP | 66 | 34000 → 60709 [FIN, ACK] Seq=1 Ack=13 Win=5792 Len=0 TStamp=125737 TSectr=125758 |
| 12 | 4.172611 | 10.0.0.2 | 11.0.0.2 | TCP | 66 | 60709 → 34000 [ACK] Seq=13 Ack=2 Win=5840 Len=0 TStamp=125758 TSectr=125737 |
| 13 | 4.989294 | de:02:94:29:fc:d4 | 8e:d5:68:dd:a2:ce | ARP | 42 | Who has 10.0.0.2? Tell 10.0.0.1 |
| 14 | 4.990194 | 8e:d5:68:dd:a2:ce | de:02:94:29:fc:d4 | ARP | 42 | 10.0.0.2 is at 8e:d5:68:dd:a2:ce |

Protocolo HTTP

Es el protocolo empleado para acceder a páginas web a través de Internet. Estudiaremos el funcionamiento general de este protocolo sin profundizar demasiado en los detalles concretos.

Ejercicio 7

Procedimiento: Análisis de Flujo TCP (Follow TCP Stream)

Para analizar la sesión web completa contenida en parte2.pcap:

- Identificación: Localiza un paquete dirigido a la IP 195.81.202.109 con puerto destino 80 (HTTP).
- Acción: Haz clic derecho sobre dicho paquete y selecciona la opción Follow > TCP Stream.
- Resultado:
 - Se abre una ventana emergente mostrando el contenido ASCII de la conversación completa.
 - Rojo: Peticiones enviadas por el cliente (nosotros).
 - Azul: Respuestas enviadas por el servidor web.
- Filtrado automático: Al cerrar la ventana emergente, Wireshark aplica automáticamente un filtro de sesión (ej: tcp.stream eq 0) para aislar esos paquetes del resto.

Identifica el primer mensaje http (después de seleccionar la opción follow TCP Stream) y obtén de este mensaje la dirección IP del destino, la versión del protocolo HTTP que se está utilizando y la dirección IP del origen.

Identifica un mensaje cuyo método solicita un archivo del servidor, estos mensajes se identifican por llevar el comando GET. Consulta el campo URI del protocolo http y especifica la dirección donde se localiza dicho archivo.

1. Identifica el primer mensaje HTTP y obtén sus datos Mirando el Paquete 43 (el primero de la lista con protocolo HTTP):

- Dirección IP del destino: 195.81.202.109 (Se ve en la cabecera IP: Dst: 195.81.202.109).
- Versión del protocolo HTTP: HTTP/1.1 (Aparece al final de la línea de solicitud: Request Version: HTTP/1.1 o en la columna Info).
- Dirección IP del origen: 192.168.117.203 (Se ve en la cabecera IP: Src: 192.168.117.203).

2. Consulta el campo URI y especifica la dirección donde se localiza dicho archivo Analizando los detalles del protocolo "Hypertext Transfer Protocol" en la parte inferior de la primera imagen:

- **Archivo solicitado:** El mensaje es un GET que pide el archivo /participa/lfm/2010-2011/flash/fotos/cambios.jpg.
- **Dirección de localización (Full request URI):** Wireshark reconstruye la dirección completa abajo del todo.
 - Respuesta: <http://estaticos.marca.com/participa/lfm/2010-2011/flash/fotos/cambios.jpg>

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|-----------------|-----------------|----------|--------|--|
| 43 | 3.317469 | 192.168.117.203 | 195.81.202.109 | HTTP | 792 | GET /participa/lfm/2010-2011/flash/fotos/cambios.jpg HTTP/1.1 |
| 44 | 3.332765 | 195.81.202.109 | 192.168.117.203 | TCP | 407 | 80 → 49716 [PSH, ACK] Seq=1 Ack=739 Win=31 Len=353 [TCP PDU reassembled in 99] |

Con esto te muestra una ventana emergente mostrando el contenido ASCII de la conversación completa.

-Rojo: Peticiones enviadas por el cliente (nosotros).

-Azul: Respuestas enviadas por el servidor web.

Protocolo TELNET

El propósito del protocolo TELNET es permitir un método estándar de comunicar entre sí terminales y procesos orientados a terminal. Su mayor problema es de seguridad, ya que todos los nombres de usuario y contraseñas necesarias para entrar en las máquinas viajan por la red como texto plano (cadenas de texto sin cifrar). Esto facilita que cualquiera que espíe el tráfico de la red pueda obtener los nombres de usuario y contraseñas.

Ejercicio 8

Para analizar la estructura de este protocolo con Wireshark vamos a abrir el fichero telnet.pcap.

Para facilitar el análisis, es recomendable aplicar un filtro de presentación indicando “telnet” para mostrar únicamente mensajes de este protocolo.

Mira la información de los paquetes que has enviado con este protocolo y determina el puerto que se utiliza para telnet. Lee el contenido de los paquetes, para ello tendrás que mirar el contenido en ASCII del campo

Data del protocolo telnet. Localiza la solicitud de login y password del servidor, así como la lectura de los datos que se enviaron, ten presente que puedes necesitar varios paquetes para obtener finalmente el login y password que se escribieron. ¿Qué nombre de usuario y qué contraseña utilizó para iniciar sesión?

1. Puerto utilizado

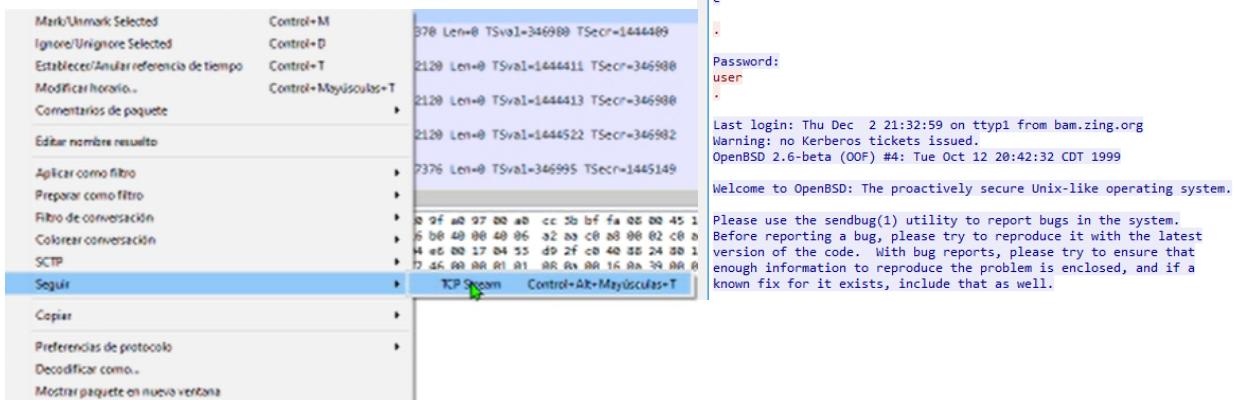
- Respuesta: Puerto 23 (TCP).

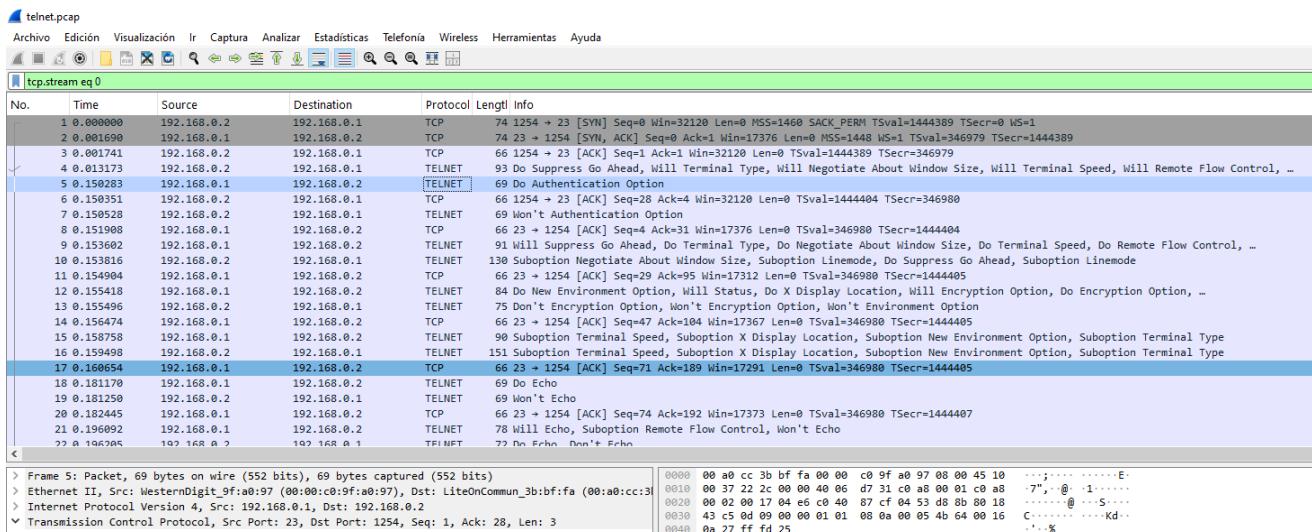
2. Nombre de usuario y contraseña Mirando el texto en azul (lo que envía el servidor) y rojo (lo que envía el cliente) en tu ventana:

- El servidor dice: login:
- El usuario responde (letra a letra): fake
- El servidor dice: Password:
- El usuario responde: user

Respuesta:

- Usuario: fake
- Contraseña: user





Protocolo SSH

SSH permite a los usuarios registrarse en sistemas de host remotamente. A diferencia de telnet, visto anteriormente, SSH encripta la sesión de registro imposibilitando que alguien pueda obtener una contraseña de texto. El uso de métodos seguros para registrarse remotamente a otros sistemas hace disminuir los riesgos de seguridad para ambos sistemas y el sistema remoto.

Ejercicio 9

Para analizar la estructura de este protocolo con Wireshark vamos a seguir los mismos pasos que para el protocolo anterior, pero en este caso abriremos el archivo ssh.pcap.

Para facilitar el análisis, es recomendable aplicar un filtro de presentación para que sólo aparezcan los paquetes de este protocolo (ssh).

Mira la información de los paquetes que has enviado con este protocolo y determina el puerto que se utiliza para ssh.

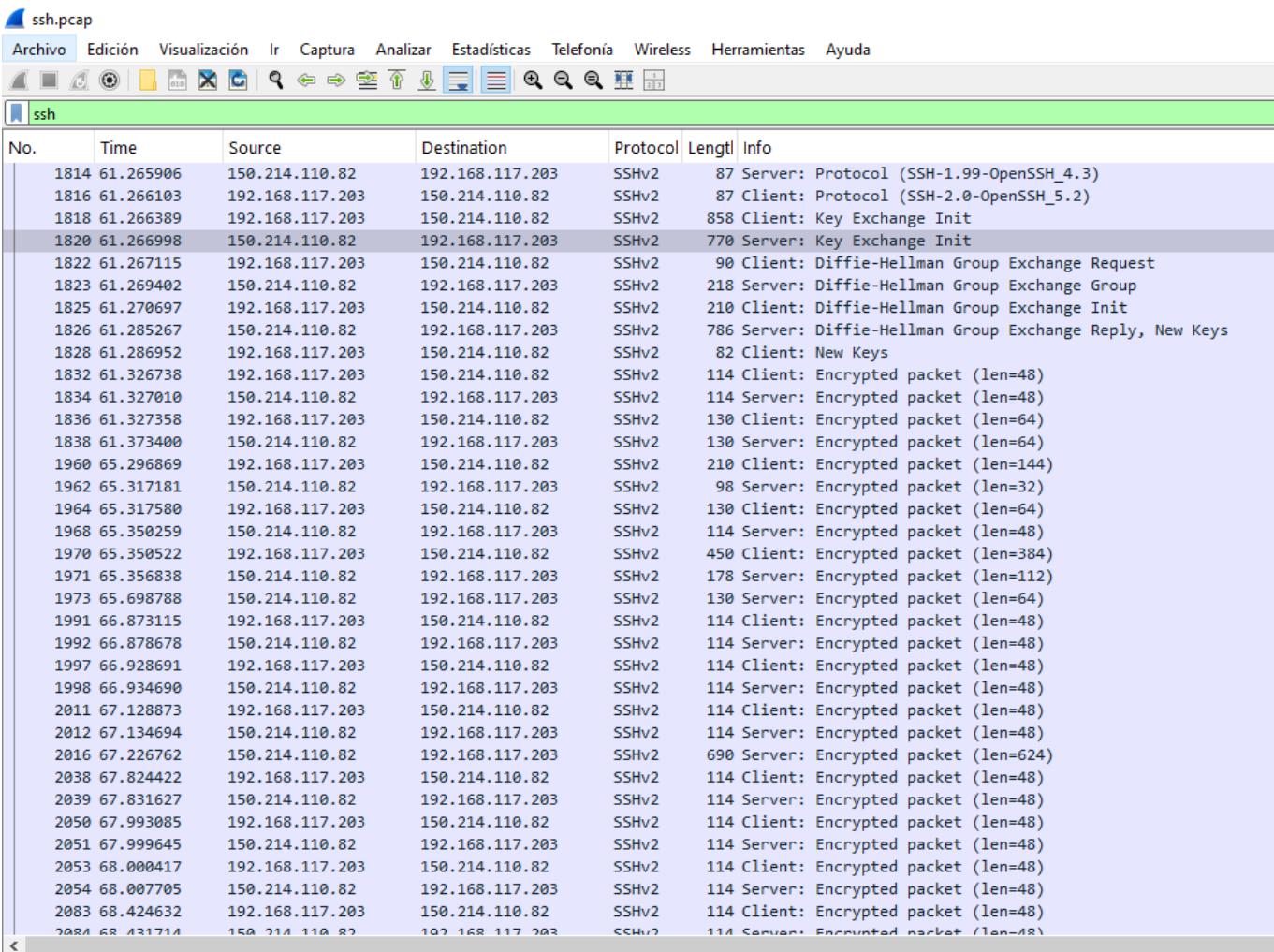
Lee el contenido de los paquetes, para ello tendrás que mirar la ventana que muestra el contenido ASCII de los campos del protocolo ssh. ¿Encuentras el login y las claves que se han introducido para iniciar la sesión remota?

1. Puerto que se utiliza para SSH Mirando el panel de detalles en tu captura principal (image_ed96f6.png):

- En la cabecera "Transmission Control Protocol", se ve claramente que el puerto del servidor (Source Port) es el 22.
- Respuesta: Puerto 22.

2. ¿Encuentras el login y las claves para iniciar la sesión? Mirando tu captura de la ventana de seguimiento (image_ed9376.png):

- A diferencia de Telnet, donde leímos "fake" y "user", aquí solo se ven caracteres extraños e ilegibles después de la versión del protocolo (SSH-2.0...).
- **Respuesta:** No, no es posible encontrar el login ni las claves.
- **Explicación:** Esto ocurre porque SSH cifra toda la comunicación (de ahí su nombre, Secure Shell). Aunque captures los paquetes, la información viaja encriptada y un atacante no puede leer tu contraseña, lo que lo hace mucho más seguro que Telnet.



```
> Frame 1820: Packet, 770 bytes on wire (6160 bits), 770 bytes captured (6160 bits)
  ✓ Ethernet II, Src: Cisco_45:c4:00 (00:15:c7:45:c4:00), Dst: Apple_14:9d:56 (c4:2c:03:14:9d:56)
    ✓ Destination: Apple_14:9d:56 (c4:2c:03:14:9d:56)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ..0. .... .... .... .... = IG bit: Individual address (unicast)
    > Source: Cisco_45:c4:00 (00:15:c7:45:c4:00)
      Type: IPv4 (0x0800)
      [Stream index: 8]
  > Internet Protocol Version 4, Src: 150.214.110.82, Dst: 192.168.117.203
  ✓ Transmission Control Protocol, Src Port: 22, Dst Port: 49748, Seq: 22, Ack: 814, Len: 704
    Source Port: 22
    Destination Port: 49748
```

| | | | | | | | | | |
|------|----|----|----|----|----|----|----|----|----|
| 0000 | c4 | 2c | 03 | 14 | 9d | 56 | 00 | 15 | c7 |
| 0010 | 02 | f4 | 12 | 37 | 40 | 00 | 3f | 06 | eb |
| 0020 | 75 | cb | 00 | 16 | c2 | 54 | 7e | 67 | 08 |
| 0030 | 00 | 74 | 93 | 80 | 00 | 00 | 01 | 01 | 08 |
| 0040 | c1 | 15 | 00 | 00 | 02 | bc | 07 | 14 | 72 |
| 0050 | 52 | e8 | f8 | f5 | 51 | 61 | 43 | 57 | 00 |
| 0060 | 69 | 65 | 2d | 68 | 65 | 6c | 6c | 6d | 61 |
| 0070 | 2d | 65 | 78 | 63 | 68 | 61 | 6e | 67 | 65 |
| 0080 | 69 | 66 | 66 | 69 | 65 | 6d | 68 | 65 | 6c |
| 0090 | 6f | 75 | 70 | 31 | 34 | 2d | 73 | 68 | 61 |
| 00a0 | 65 | 2d | 68 | 65 | 6c | 6c | 6d | 61 | 6e |
| 00b0 | 2d | 73 | 68 | 61 | 31 | 00 | 00 | 00 | 0f |
| 00c0 | 2c | 73 | 73 | 68 | 2d | 64 | 73 | 73 | 00 |

