

Práctica 1: Entorno de trabajo auditoría de redes inalámbricas

Seguridad de las Comunicaciones Inalámbricas

Autor: Cheuk Kelly Ng Pante (alu0101364544@ull.edu.es)

Fecha: 8 de diciembre de 2025

Índice general

1. Análisis y Configuración de Interfaces de Red	1
1.1. Enumeración de Interfaces	1
2. Gestión del Estado de la Interfaz y Escalado de Privilegios	2
2.1. Intento sin privilegios	2
2.2. Ejecución con privilegios	2
2.3. Verificación de conectividad	3
2.4. Reactivación	4
2.4.1. Comandos adicionales	4
3. Verificación de la Pila TCP/IP y Conectividad Externa	5
3.1. Prueba de Loopback	5
3.2. Prueba de Conectividad Externa	6
4. Análisis de la Caché del Protocolo ARP	6
5. Inspección de la Tabla de Enrutamiento	7
6. Trazado de Ruta de Red (Traceroute)	7
7. Escaneo de Puertos Locales con Nmap	8
8. Análisis de Sockets y Conexiones de Red	9
9. Resolución de Nombres DNS (NSLookup)	10
9.1. Objetivo	10
10. Interacción de Red con Netcat (nc)	10
10.1. Sintaxis clave	11
11. Reconocimiento DNS Avanzado con dnsenum	11
12. Análisis de Tráfico en Terminal con tcpdump	13

1. Análisis y Configuración de Interfaces de Red

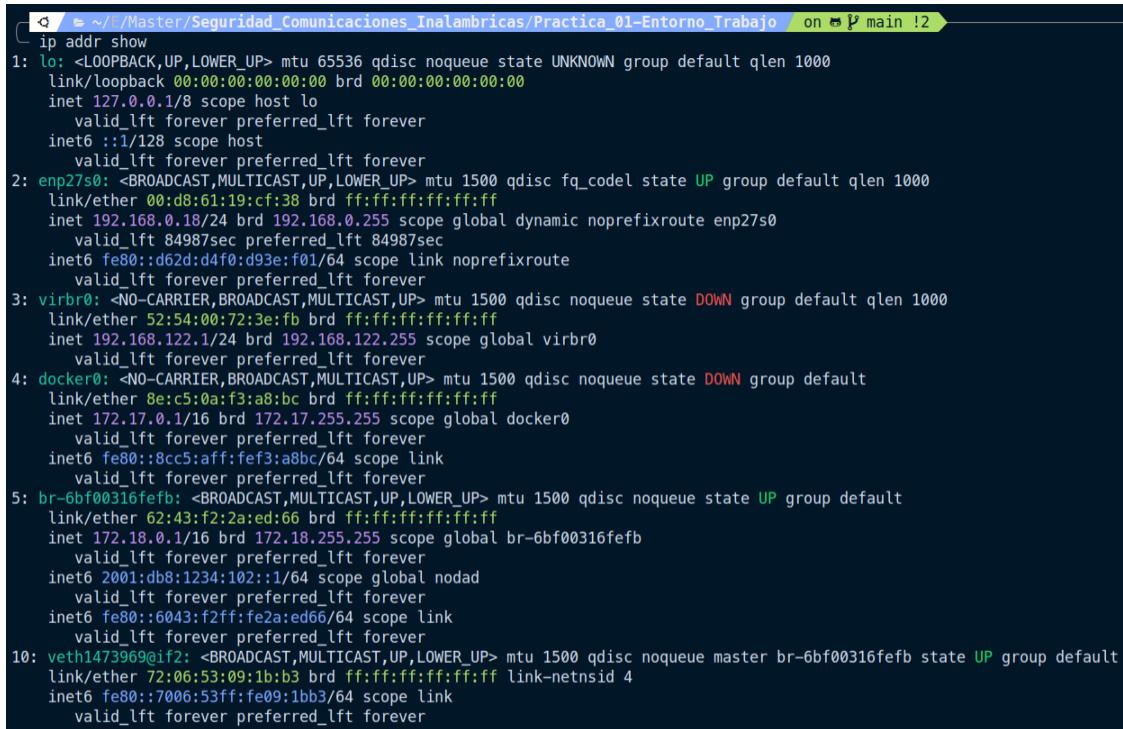
Ejecutar el comando para listar todas las interfaces de red:

```
1 ip addr show
```

Forma abreviada: ip a

1.1. Enumeración de Interfaces

- **Dirección MAC:** Valor link/ether
- **Direcciones IP:** inet (IPv4) e inet6 (IPv6)
- **Dirección Broadcast:** Valor brd
- **Estado:** STATE UP o STATE DOWN
- **MTU:** Valor numérico mtu



```
ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: enp27s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:d8:61:19:cf:38 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.18/24 brd 192.168.0.255 scope global dynamic noprefixroute enp27s0
        valid_lft 84987sec preferred_lft 84987sec
        inet6 fe80::d62d:d4f0:d93e:101/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
3: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 52:54:00:72:3e:fb brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
        valid_lft forever preferred_lft forever
4: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 8e:c5:0a:f3:a8:bc brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
        inet6 fe80::8c5:aff:fef3:a8bc/64 scope link
            valid_lft forever preferred_lft forever
5: br-6bf00316fefb: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 62:43:f2:2a:ed:66 brd ff:ff:ff:ff:ff:ff
    inet 172.18.0.1/16 brd 172.18.255.255 scope global br-6bf00316fefb
        valid_lft forever preferred_lft forever
        inet6 2001:db8:1234:102::1/64 scope global nodad
            valid_lft forever preferred_lft forever
        inet6 fe80::6043:f2ff:fe2a:ed66/64 scope link
            valid_lft forever preferred_lft forever
10: veth1473969@if2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-6bf00316fefb state UP group default
    link/ether 72:06:53:09:1b:b3 brd ff:ff:ff:ff:ff:ff link-netnsid 4
    inet6 fe80::7006:53ff:fe09:1bb3/64 scope link
        valid_lft forever preferred_lft forever
```

Figura 1.1: Salida del comando ip addr show

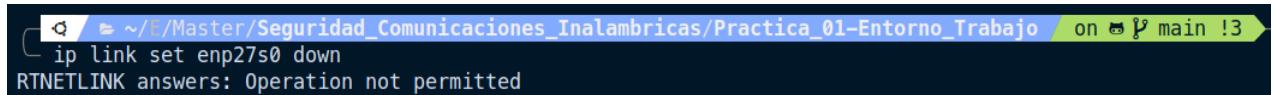
2. Gestión del Estado de la Interfaz y Escalado de Privilegios

2.1. Intento sin privilegios

Intentar deshabilitar la interfaz eth0:

```
1 ip link set eth0 down
```

Resultado esperado: Error de permiso denegado.



A terminal window showing a command being run. The command is "ip link set eth0 down". The output shows an error message: "RTNETLINK answers: Operation not permitted".

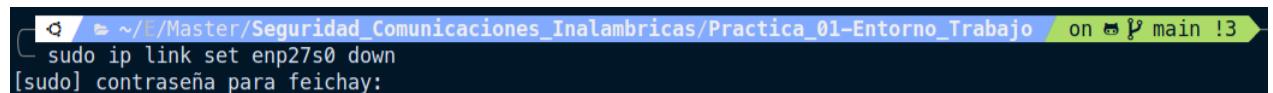
```
ip link set eth0 down
RTNETLINK answers: Operation not permitted
```

Figura 2.1: Error al intentar deshabilitar eth0 sin privilegios

2.2. Ejecución con privilegios

Ejecutar con sudo:

```
1 sudo ip link set eth0 down
```



A terminal window showing a command being run with sudo. The command is "sudo ip link set eth0 down". A password prompt "[sudo] contraseña para feichay:" is visible.

```
sudo ip link set eth0 down
[sudo] contraseña para feichay:
```

Figura 2.2: Deshabilitación exitosa de eth0 con privilegios

2.3. Verificación de conectividad

Confirmar que la interfaz está inactiva:

```
1 ip a
```

```
1 ~ ~/Master/Seguridad_Comunicaciones_Inalambricas/Practica_01-Entorno_Trabajo on m p main !5
  sudo ip link set enp27s0 down
2 ~ ~/Master/Seguridad_Comunicaciones_Inalambricas/Practica_01-Entorno_Trabajo on m p main !5
  ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00 brd 00:00:00:00:00
      inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
      inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp27s0: <BROADCAST,MULTICAST> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
    link/ether 00:d8:61:19:cf:38 brd ffffff:ffff:ffff:ffff
      inet 192.168.0.18/24 brd 192.168.0.255 scope global dynamic noprefixroute enp27s0
        valid_lft 8630/sec preferred_lft 8630/sec
        inet6 fe80::94ee:19ff:fe00:18%enp27s0/64 scope link tentative noprefixroute
          valid_lft forever preferred_lft forever
3: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 52:54:00:72:3e:fb brd ffffff:ffff:ffff:ffff
      inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
        valid_lft forever preferred_lft forever
4: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 0e:c5:0a:f3:a8:bc brd ffffff:ffff:ffff:ffff
      valid_lft forever preferred_lft forever
      inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
      inet6 fe80::8cc:aff:fe:f3:a8bc/64 scope link
        valid_lft forever preferred_lft forever
5: br-6bf00316fefb: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 62:43:f2:2a:ed:66 brd ffffff:ffff:ffff:ffff
      inet 172.18.0.1/16 brd 172.18.255.255 scope global br-6bf00316fefb
        valid_lft forever preferred_lft forever
      inet6 2001:db8:1234:102::1/64 scope global nodad
        valid_lft forever preferred_lft forever
      inet6 fe80::6043:f2ff:fe2a:ed66/64 scope link
        valid_lft forever preferred_lft forever
10: veth1473999qif2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-6bf00316fefb state UP group default
    link/ether 72:06:53:09:1b:b3 brd ffffff:ffff:ffff link-netnsid 4
      inet6 fe80::7006:53ff:fe09:1bb3/64 scope link
        valid_lft forever preferred_lft forever
```

Figura 2.3: Verificación de estado inactivo de eth0

Al acceder a alguna página web podemos ver que se ha perdido la conexión:

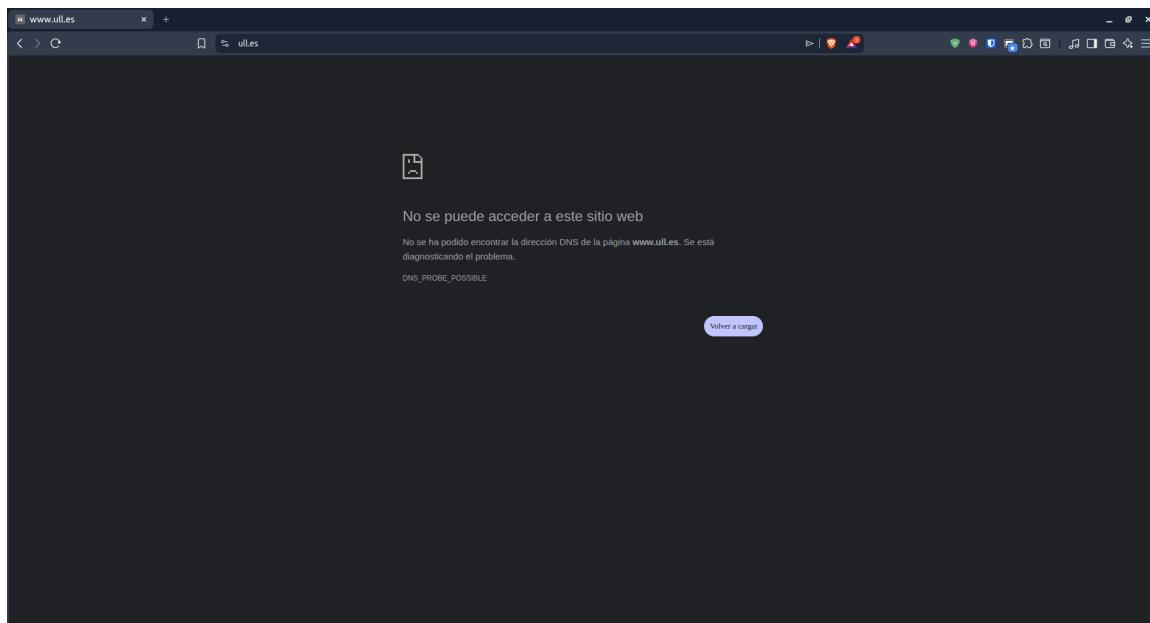


Figura 2.4: Perdida de conexión

2.4. Reactivación

Restaurar la interfaz:

```
1 sudo ip link set eth0 up
```

Validar la restauración con ip a y verificar conectividad a Internet.

The terminal window shows the command 'sudo ip link set eth0 up' being run. It then displays the output of 'ip a' which lists various network interfaces including 'lo', 'enp27s0', 'virbr0', 'docker0', and several bridge interfaces ('br-...', 'veth...', 'veth147396901f2'). Finally, a 'ping 8.8.8.8' command is run, showing four successful packets transmitted to Google's public DNS server.

```
1 ~ /~/Seguridad_Comunicaciones_Inalambricas/Practica_01-Entorno_Trabajo on ✘ p main 16 ?1
2 sudo ip link set enp27s0 up
3 ip a
4 lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
5 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
6     inet 127.0.0.1/8 scope host lo
7         valid_lft forever preferred_lft forever
8     inet6 ::1/128 scope host
9         valid_lft forever preferred_lft forever
10    2: enp27s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
11     link/ether 00:08:61:19:c3:38 brd ff:ff:ff:ff:ff:ff
12     inet 192.168.0.18/24 brd 192.168.0.255 scope global dynamic noprefixroute enp27s0
13         valid_lft 86277sec preferred_lft 86277sec
14     inet6 fe80::0861:19ff:fe:c338/64 scope link noprefixroute
15         valid_lft forever preferred_lft forever
16     virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
17     link/ether 52:54:00:72:3e:fb brd ff:ff:ff:ff:ff:ff
18     inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
19         valid_lft forever preferred_lft forever
20     docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
21     link/ether 8e:c5:0a:f3:a8:bc brd ff:ff:ff:ff:ff:ff
22     inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
23         valid_lft forever preferred_lft forever
24     inet6 fe80::cc5:aff:fef3:a8bc/64 scope link
25         valid_lft forever preferred_lft forever
26     5: br-6bf00316fefb: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
27     link/ether 62:43:f2:2a:ed:66 brd ff:ff:ff:ff:ff:ff
28     inet 172.18.0.1/16 brd 172.18.255.255 scope global br-6bf00316fefb
29         valid_lft forever preferred_lft forever
30     inet6 2001:db8:1234:102::1/64 scope global nodad
31         valid_lft forever preferred_lft forever
32     inet6 fe80::6043:f2ff:fe2a:ed66/64 scope link
33         valid_lft forever preferred_lft forever
34     veth147396901f2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-6bf00316fefb state UP group default
35     link/ether 72:06:53:09:1b:b3 brd ff:ff:ff:ff:ff:ff link-netnsid 4
36     inet6 fe80::7006:53ff:fe09:1bb3/64 scope link
37         valid_lft forever preferred_lft forever
38
39 1 ~ /~/Master/Seguridad_Comunicaciones_Inalambricas/Practica_01-Entorno_Trabajo on ✘ p main 16 ?1
40 ping 8.8.8.8 -c 4
41 PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
42 64 bytes from 8.8.8.8: icmp_seq=1 ttl=19 time=29.0 ms
43 64 bytes from 8.8.8.8: icmp_seq=2 ttl=19 time=29.0 ms
44 64 bytes from 8.8.8.8: icmp_seq=3 ttl=19 time=29.0 ms
45 64 bytes from 8.8.8.8: icmp_seq=4 ttl=19 time=30.3 ms
46
47 --- 8.8.8.8 ping statistics ---
48 4 packets transmitted, 4 received, 0% packet loss, time 3003ms
49 rtt min/avg/max/mdev = 28.993/29.329/30.254/0.534 ms
```

Figura 2.5: Reactivación exitosa de eth0

2.4.1. Comandos adicionales

```
1 sudo ip addr add 192.168.1.200/24 dev eth0
```

The terminal window shows the command 'sudo ip addr add 192.168.1.200/24 dev enp27s0' being run. It then displays the output of 'ip a' which lists the same interfaces as Figure 2.5, but with the addition of the new IP configuration for 'enp27s0'.

```
1 ~ /~/Master/Seguridad_Comunicaciones_Inalambricas/Practica_01-Entorno_Trabajo on ✘ p main !7 ?1
2 sudo ip addr add 192.168.1.200/24 dev enp27s0
3 ip a
4 lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
5 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
6     inet 127.0.0.1/8 scope host lo
7         valid_lft forever preferred_lft forever
8     inet6 ::1/128 scope host
9         valid_lft forever preferred_lft forever
10    2: enp27s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
11     link/ether 00:08:61:19:c3:38 brd ff:ff:ff:ff:ff:ff
12     inet 192.168.0.18/24 brd 192.168.0.255 scope global dynamic noprefixroute enp27s0
13         valid_lft 86184sec preferred_lft 86184sec
14     inet 192.168.1.200/24 brd 192.168.1.255 scope global enp27s0
15         valid_lft forever preferred_lft forever
16     inet6 fe80::0861:19ff:fe:c338/64 scope link noprefixroute
17         valid_lft forever preferred_lft forever
```

Figura 2.6: Asignación temporal de nueva IP a eth0

Al hacer esto el cambio es temporal y se pierde al reiniciar. Para hacerlo persistente hay que editar el fichero `/etc/netplan/01-network-manager-all.yaml`.

```
cd ~//Seguridad_Comunicaciones_Inalambricas/Practica_01-Entorno_Trabajo
sudo vi /etc/netplan/01-network-manager-all.yaml

sudo netplan apply

# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    enp27s0:
      addresses:
        - 192.168.0.200/24
      dhcp4: false
      routes:
        - to: default
          via: 192.168.1.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]

ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
  link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
  inet 127.0.0.1/8 scope host lo
    valid_lft forever preferred_lft forever
  inet6 ::1/128 scope host
    valid_lft forever preferred_lft forever
2: enp27s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
  link/ether 00:08:61:19:c1:38 brd ff:ff:ff:ff:ff:ff
  inet 192.168.0.200/24 brd 192.168.0.255 scope global noprefixroute enp27s0
    valid_lft forever preferred_lft forever
  inet6 fe80::208:61ff:fe19:c138/64 scope link
    valid_lft forever preferred_lft forever
```

Figura 2.7: Verificación de nueva IP asignada a eth0

3. Verificación de la Pila TCP/IP y Conectividad Externa

3.1. Prueba de Loopback

Ping a la interfaz de loopback:

```
1 ping -c 4 127.0.0.1
```

```
ping -c 4 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.040 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.035 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.036 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.046 ms

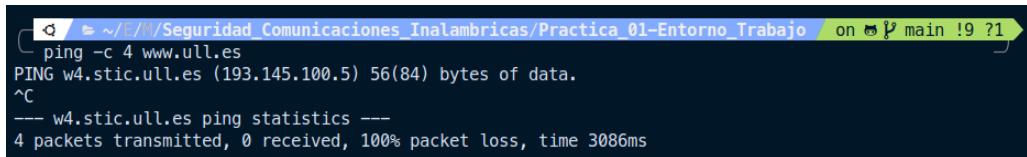
--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3086ms
rtt min/avg/max/mdev = 0.035/0.039/0.046/0.004 ms
```

Figura 3.1: Prueba de conectividad a loopback

3.2. Prueba de Conectividad Externa

Verificar resolución DNS y conectividad WAN:

```
1 ping -c 4 www.ull.es
```



```
ping -c 4 www.ull.es
PING w4.stic.ull.es (193.145.100.5) 56(84) bytes of data.
^C
--- w4.stic.ull.es ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3086ms
```

Figura 3.2: Prueba de conectividad a www.ull.es

A continuación se documentan los resultados obtenidos tras ejecutar el comando de diagnóstico de red hacia el dominio de la Universidad de La Laguna.

a. Estadísticas de Paquetes Basado en la línea final: *4 packets transmitted, 0 received, 100 % packet loss.*

- **Transmitidos:** 4
- **Recibidos:** 0
- **Pérdida (%):** 100 %

b. Estadísticas de RTT (Round-Trip Time) Debido a que la pérdida de paquetes fue total (ningún paquete retornó), el sistema no pudo calcular los tiempos de viaje.

- **Mínimo:** N/A (No disponible)
- **Promedio (Avg):** N/A (No disponible)
- **Máximo:** N/A (No disponible)

Observación: El fallo en la recepción de paquetes (100 % de pérdida) sugiere que el host destino (193.145.100.5) está inactivo o, lo más probable, que existe un firewall bloqueando las solicitudes ICMP.

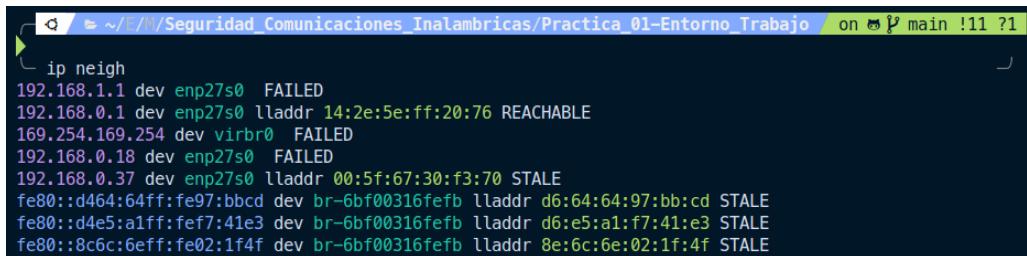
4. Análisis de la Caché del Protocolo ARP

Al realizar la práctica en Ubuntu 22.04, se observó que el comando clásico arp -a no está disponible por defecto, ya que pertenece al paquete obsoleto net-tools. Por este motivo, se utiliza la suite moderna iproute2, donde el comando ip neigh (abreviatura de ip neighbor) permite consultar y gestionar la tabla de vecinos, que para IPv4 corresponde a la caché ARP.

En consecuencia, en este informe se emplea el comando ip neigh show como alternativa directa a arp -a para inspeccionar el mapeo entre direcciones IP y direcciones MAC de los dispositivos de la red, incluida la puerta de enlace.

El comando utilizado es:

```
1 ip neigh
```



```
ip neigh
192.168.1.1 dev enp27s0 FAILED
192.168.0.1 dev enp27s0 lladdr 14:2e:5e:ff:20:76 REACHABLE
169.254.169.254 dev virbr0 FAILED
192.168.0.18 dev enp27s0 FAILED
192.168.0.37 dev enp27s0 lladdr 00:5f:67:30:f3:70 STALE
fe80::d464:64ff:fe97:bbcd dev br-6bf00316fefb lladdr d6:64:64:97:bb:cd STALE
fe80::d4e5:a1ff:fef7:41e3 dev br-6bf00316fefb lladdr d6:e5:a1:f7:41:e3 STALE
fe80::8c6c:6eff:fe02:1f4f dev br-6bf00316fefb lladdr 8e:6c:6e:02:1f:4f STALE
```

Figura 4.1: Salida del comando ip neigh

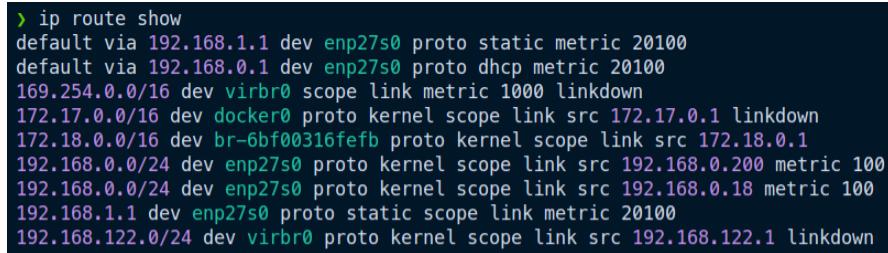
5. Inspección de la Tabla de Enrutamiento

Analizar cómo el sistema decide dónde enviar el tráfico. Para mostrar la tabla de enrutamiento:

```
1 ip route show
```

Forma abreviada: ip r

En la salida del comando ip route show se observan dos rutas por defecto configuradas en la interfaz `enp27s0`. Estas rutas apuntan a dos puertas de enlace distintas, `192.168.1.1` (configurada de forma estática) y `192.168.0.1` (obtenida por DHCP), ambas con la misma métrica (20100), por lo que el kernel puede utilizar cualquiera de ellas como *default gateway* en función de su disponibilidad.



```
ip route show
default via 192.168.1.1 dev enp27s0 proto static metric 20100
default via 192.168.0.1 dev enp27s0 proto dhcp metric 20100
169.254.0.0/16 dev virbr0 scope link metric 1000 linkdown
172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown
172.18.0.0/16 dev br-6bf00316fefb proto kernel scope link src 172.18.0.1
192.168.0.0/24 dev enp27s0 proto kernel scope link src 192.168.0.200 metric 100
192.168.0.0/24 dev enp27s0 proto kernel scope link src 192.168.0.18 metric 100
192.168.1.1 dev enp27s0 proto static scope link metric 20100
192.168.122.0/24 dev virbr0 proto kernel scope link src 192.168.122.1 linkdown
```

Figura 5.1: Salida del comando ip route show

6. Trazado de Ruta de Red (Traceroute)

Los trazados de ruta realizados son:

```
1 traceroute www.ull.es
2 traceroute www.net.berkeley.edu
```

```
> traceroute www.ull.es
traceroute to www.ull.es (193.145.100.5), 30 hops max, 60 byte packets
 1 _gateway (192.168.0.1)  3.405 ms  3.362 ms  3.407 ms
 2 * * *
 3 * * *
 4 rediris.baja.espanix.net (193.149.1.26)  32.335 ms  32.316 ms  32.296 ms
 5 ciemat-rt2.ethtrunk1-315.iac.rt2.can.red.rediris.es (130.206.245.62)  61.405 ms  61.384
ms 61.366 ms
 6 * * *
 7 * * *
 8 * * *
 9 * * *
10 * * *
11 * * *
12 * * *
13 * * *
14 * * *
15 * * *
16 * * *
17 * * *
18 * * *
19 * * *
20 * * *
21 * * *
22 * * *
23 * * *
24 * * *
25 * * *
26 * * *
27 * * *
28 * * *
29 * * *
30 * * *
```

(a) Trazado de ruta a www.ull.es

```
> traceroute www.net.berkeley.edu
traceroute to www.net.berkeley.edu (128.32.206.188), 30 hops max, 60 byte packets
 1 _gateway (192.168.0.1)  3.144 ms  3.168 ms  3.148 ms
 2 * * *
 3 * * *
 4 ae14-xcr1.max.cw.net (195.2.4.65)  30.838 ms  30.807 ms  30.837 ms
 5 ae6-xcr2.nyk.cw.net (195.2.2.101)  110.345 ms  110.332 ms  110.314 ms
 6 lumen-3356.nyk.cw.net (195.2.12.114)  110.403 ms *  108.056 ms
 7 * * *
 8 CENIC.edge9.SanJose1.Level3.net (4.15.122.46)  172.497 ms  171.119 ms  171.082 ms
 9 emv11-agg-01--svl-agg10--400g-01.cenic.net (137.164.11.95)  170.964 ms  173.364 ms  170
,369 ms
10 ucb-emv11-agg-01--100g.cenic.net (137.164.3.27)  170.834 ms  172.445 ms  170.702 ms
11 sut-mdc-cr1--et-0-0-1.net.berkeley.edu (128.32.0.37)  172.348 ms reccev-cev-cr1--et-0-0-
1.net.berkeley.edu (128.32.0.39)  170.098 ms  170.641 ms
12 reccev-cev-cr2--et-0-2-9.net.berkeley.edu (128.32.255.43)  172.411 ms  170.628 ms sut-md
c-cr2--et-0-2-9.net.berkeley.edu (128.32.255.175)  172.259 ms
13 ewdc-322-dr2--et54-1.net.berkeley.edu (128.32.255.45)  172.183 ms ewdc-322-dr3--et53-1.n
et.berkeley.edu (128.32.255.51)  172.171 ms  173.686 ms
14 redirect.net.berkeley.edu (128.32.206.188)  173.572 ms  171.970 ms  170.857 ms
```

(b) Trazado de ruta a www.net.berkeley.edu

Figura 6.1: Trazados de ruta a dos destinos diferentes

Al realizar la práctica en Ubuntu 22.04 se ejecutó el comando `traceroute` hacia dos destinos: `www.net.berkeley.edu` y `www.ull.es`, con el objetivo de identificar los routers intermedios y la latencia asociada a cada salto de Capa 3. En el caso de `www.net.berkeley.edu`, el trazado muestra un total de 14 saltos desde la puerta de enlace local (192.168.0.1), con un tiempo de ida y vuelta inicial de aproximadamente 3 ms y latencias que aumentan progresivamente hasta valores cercanos a 170–180 ms en el destino final, lo que refleja la distancia geográfica y el número de dominios de red atravesados.

Para `www.ull.es`, el traceroute alcanza la puerta de enlace local y varios routers pertenecientes a la red académica (por ejemplo, nodos de RedIRIS), con latencias en torno a 30–60 ms. A partir de cierto punto, numerosos saltos aparecen como `* * *`, lo que indica que esos routers no responden a las sondas de `traceroute` (por filtrado o limitación de tráfico ICMP), aun cuando la conectividad con el destino sigue siendo funcional.

7. Escaneo de Puertos Locales con Nmap

Para realizar el auto-descubrimiento de servicios se ejecutó un escaneo con `nmap` sobre la propia máquina (`nmap -sV localhost`). Este comando permite identificar qué puertos TCP se encuentran en estado `LISTEN` y, además, intenta determinar la versión de los servicios asociados mediante el flag `-sV`, lo que simula un reconocimiento inicial desde la perspectiva de un atacante situado en la misma red local.

```

> nmap -sV localhost
Starting Nmap 7.80 ( https://nmap.org ) at 2025-12-07 12:31 WET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00021s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
631/tcp    open  ipp      CUPS 2.4

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
.

Nmap done: 1 IP address (1 host up) scanned in 6.38 seconds

```

Figura 7.1: Salida del comando `nmap -sV localhost`

Al ejecutar el comando `nmap -sV localhost` sobre la propia máquina, se comprobó que el host `localhost` (127.0.0.1) está activo y responde con una latencia muy baja. El escaneo analiza los 1000 puertos TCP más habituales y muestra que 999 de ellos se encuentran cerrados, detectándose únicamente un puerto abierto: el 631/tcp, asociado al servicio `ipp` (Internet Printing Protocol), que en este caso corresponde a CUPS 2.4, es decir, el subsistema de impresión del sistema operativo escuchando en la interfaz de *loopback*.

8. Análisis de Sockets y Conexiones de Red

Para inspeccionar los sockets y conexiones de red activas en el sistema, se utilizó el comando `ss` (socket statistics), que forma parte de la suite `iproute2` y ofrece una visión detallada de las conexiones TCP, UDP y otros tipos de sockets.

```
1 ss -tuln
```

```

> ss -tuln
Netid  State  Recv-Q  Send-Q  Local Address:Port      Peer Address:Port  Process
udp    UNCONN  0        0        0.0.0.0:59988        0.0.0.0:*
udp    UNCONN  0        0        224.0.0.251:5353       0.0.0.0:*
udp    UNCONN  0        0        224.0.0.251:5353       0.0.0.0:*
udp    UNCONN  0        0        0.0.0.0:5353          0.0.0.0:*
udp    UNCONN  0        0        192.168.122.1:53       0.0.0.0:*
udp    UNCONN  0        0        127.0.0.53%lo:53       0.0.0.0:*
udp    UNCONN  0        0        0.0.0.0@virbr0:67       0.0.0.0:*
udp    UNCONN  0        0        [::]:52399           [::]:*
udp    UNCONN  0        0        [::]:5353            [::]:*
tcp    LISTEN  0        4096    127.0.0.53%lo:53       0.0.0.0:*
tcp    LISTEN  0        128     127.0.0.1:631         0.0.0.0:*
tcp    LISTEN  0        32      192.168.122.1:53       0.0.0.0:*
tcp    LISTEN  0        511     127.0.0.1:40563        0.0.0.0:*
tcp    LISTEN  0        511     127.0.0.1:6463         0.0.0.0:*
tcp    LISTEN  0        128    [::]:631              [::]:*

```

Figura 8.1: Salida del comando `ss -tuln`

Al ejecutar el comando `ss -tuln` se obtuvo un listado de los sockets UDP y TCP abiertos en el sistema, incluyendo su estado y la dirección/puerto local asociados. En la parte UDP, los sockets aparecen en estado `UNCONN`, con varios puertos 53 y 5353 escuchando en direcciones como `127.0.0.53%lo`, `192.168.122.1` y `0.0.0.0`, lo que indica la presencia de servicios de resolución de nombres (DNS) y de descubrimiento en la red local, así como un servidor DHCP vinculado a la interfaz virtual `virbr0`.

En la sección TCP, los sockets se muestran en estado `LISTEN`, destacando que la mayoría de servicios están restringidos a la interfaz de *loopback* (por ejemplo, `127.0.0.53%lo:53`, `127.0.0.1:631` y `[::1]:631`, correspondientes a servicios de DNS local y CUPS), mientras que otros puertos como `192.168.122.1:53` sólo son accesibles desde la red virtual interna. En conjunto, la salida evidencia que los servicios expuestos

hacia el exterior son mínimos y que los puertos más sensibles se limitan a la propia máquina o a redes virtuales controladas.

9. Resolución de Nombres DNS (NSLookup)

9.1. Objetivo

Realizar consultas DNS para resolver nombres de dominio a `www.ull.es` y `www.w3c.org` se utilizaron consultas DNS mediante `nslookup`. En ambos casos se identificó el servidor DNS que resolvió la consulta (línea `Server:`) y se registraron las direcciones IPv4 devueltas como registros de tipo A para cada dominio.

```
> nslookup www.ull.es
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
www.ull.es      canonical name = w4.stic.ull.es.
Name:   w4.stic.ull.es
Address: 193.145.100.5

> nslookup www.w3c.org
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
www.w3c.org      canonical name = webredir.vip.gandi.net.
Name:   webredir.vip.gandi.net
Address: 217.70.184.50
```

Figura 9.1: Salida del comando `nslookup <dominio>`

En las consultas DNS realizadas con `nslookup` para `www.ull.es` y `www.w3c.org`, el servidor que resuelve ambas peticiones es `127.0.0.53`, es decir, el resolvedor local de Ubuntu que actúa como caché y reenvía las consultas a los DNS configurados en el sistema.

Para `www.ull.es`, la respuesta incluye un nombre canónico `w4.stic.ull.es` y un registro A con la dirección IPv4 `193.145.100.5`. Para `www.w3c.org`, el nombre canónico resuelto es `webredir.vip.gandi.net` y el registro A obtenido es la dirección IPv4 `217.70.184.50`

10. Interacción de Red con Netcat (nc)

Se utilizó Netcat (`nc`) como herramienta de depuración de red. En modo escucha se empleó la sintaxis `nc -l -p 1234` para iniciar un servicio que acepta conexiones entrantes en el puerto 1234. En modo cliente se utilizó `nc <host>1234` para establecer una conexión TCP hacia dicho puerto remoto y verificar el intercambio de datos extremo a extremo.

10.1. Sintaxis clave

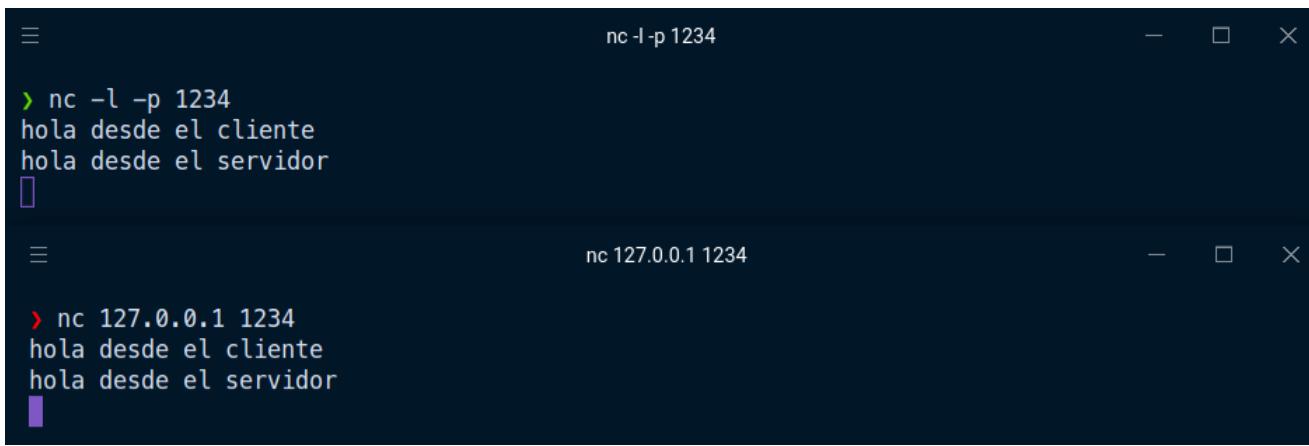
Modo Escucha (Listener):

```
1 nc -l -p 1234
```

Modo Cliente:

```
1 nc [IP_remota] [puerto]
```

En la Figura 10.1 se muestra la configuración de Netcat en modo escucha y cliente en dos terminales diferentes.



```
nc -l -p 1234
hola desde el cliente
hola desde el servidor
[blank]

nc 127.0.0.1 1234
hola desde el cliente
hola desde el servidor
```

Figura 10.1: Uso de Netcat en modo escucha y cliente

Para probar la interacción de red con Netcat se lanzó un listener en la máquina local mediante `nc -l -p 1234` y, en una segunda terminal, un cliente con `nc 127.0.0.1 1234`. Una vez establecida la conexión, cualquier texto introducido en una de las terminales aparecía en la otra, demostrando que se había creado un canal TCP bidireccional entre cliente y servidor sobre el puerto 1234 de `localhost`, lo que simula el funcionamiento básico de un servicio y su cliente en la misma red.

11. Reconocimiento DNS Avanzado con dnsenum

Se ejecutó `dnsenum tecnomobile.com` para realizar enumeración DNS automatizada del dominio, obteniendo direcciones de host (registros A/AAAA), servidores de nombres (NS), servidores de correo (MX) y, adicionalmente, posibles subdominios y rangos de red asociados descubiertos por fuerza bruta y consultas complementarias.

```

> dnsenum tecnomobile.com
dnsenum VERSION:1.2.6

----- tecnomobile.com -----

Host's addresses:
-----
tecnomobile.com. 600 IN A 37.48.77.81

Wildcard detection using: rdclobrxxezo
-----
rdclobrxxezo.tecnomobile.com. 600 IN A 77.247.183.146

!!!!!!!!

Wildcards detected, all subdomains will point to the same IP address
Omitting results containing 77.247.183.146.
Maybe you are using OpenDNS servers.

!!!!!!!!

Name Servers:
-----
ns1.quokkadns.com. 300 IN A 207.244.71.177
ns2.quokkadns.com. 300 IN A 5.79.65.14

Mail (MX) Servers:
-----
Trying Zone Transfers and getting Bind Versions:
-----

Trying Zone Transfer for tecnomobile.com on ns1.quokkadns.com ...
AXFR record query failed: Connection timed out

----- Trying Zone Transfers and getting Bind Versions: -----
----- Trying Zone Transfer for tecnomobile.com on ns1.quokkadns.com ...
AXFR record query failed: No route to host
----- Brute forcing with /usr/share/dnsenum/dns.txt: -----
----- 1003.tecnomobile.com. 600 IN A 37.48.77.81
1025.tecnomobile.com. 600 IN A 37.48.77.81
1027.tecnomobile.com. 600 IN A 208.115.249.234
1029.tecnomobile.com. 600 IN A 212.32.255.111
1037.tecnomobile.com. 600 IN A 37.48.65.145
1044.tecnomobile.com. 600 IN A 37.48.65.155
1066.tecnomobile.com. 600 IN A 23.82.16.54
1075.tecnomobile.com. 600 IN A 37.48.65.145
1082.tecnomobile.com. 600 IN A 37.48.77.81
11.tecnomobile.com. 600 IN A 37.48.77.81
1106.tecnomobile.com. 600 IN A 208.115.249.234
1107.tecnomobile.com. 600 IN A 172.241.213.96
1108.tecnomobile.com. 600 IN A 37.48.77.81
1114.tecnomobile.com. 600 IN A 216.245.214.86
1115.tecnomobile.com. 600 IN A 212.32.255.111
1116.tecnomobile.com. 600 IN A 23.82.16.54
112sos.tecnomobile.com. 600 IN A 37.48.65.145
1167.tecnomobile.com. 600 IN A 37.48.65.155
1168.tecnomobile.com. 600 IN A 212.32.255.111
1178.tecnomobile.com. 600 IN A 212.32.255.111
1184.tecnomobile.com. 600 IN A 37.48.65.155
1187.tecnomobile.com. 600 IN A 208.115.249.234
1189.tecnomobile.com. 600 IN A 37.48.77.81
1198.tecnomobile.com. 600 IN A 208.115.249.234
1203.tecnomobile.com. 600 IN A 192.157.56.139
1204.tecnomobile.com. 600 IN A 37.48.65.155
121.tecnomobile.com. 600 IN A 208.115.249.234
1211.tecnomobile.com. 600 IN A 208.115.249.234
1216.tecnomobile.com. 600 IN A 212.32.255.111
131.tecnomobile.com. 600 IN A 37.48.65.155
132.tecnomobile.com. 600 IN A 212.32.255.111
134.tecnomobile.com. 600 IN A 37.48.77.81
154.tecnomobile.com. 600 IN A 212.32.255.111
155.tecnomobile.com. 600 IN A 37.48.65.145

```

Figura 11.1: Salida del comando `dnsenum tecnomobile.com`

Mediante la ejecución de `dnsenum tecnomobile.com` se identificó que el dominio principal `tecnomobile.com` dispone de un registro A asociado a la dirección IPv4 `37.48.77.81`. Asimismo, se detectaron como servidores de nombres (registros NS) los hosts `ns1.quokkadns.com` (`207.244.71.177`) y `ns2.quokkadns.com` (`5.79.65.14`). La herramienta no devolvió registros MX para el dominio, pero sí enumeró varios subdominios numéricos (por ejemplo, `1003.tecnomobile.com`, `1025.tecnomobile.com`, etc.) con distintas direcciones IPv4, lo que aporta información adicional sobre la infraestructura asociada al dominio.

12. Análisis de Tráfico en Terminal con tcpdump

Se realizó una captura básica con `tcpdump` mediante el comando `sudo tcpdump -i enp27s0 -n -c 20`. En las primeras líneas se observa tráfico de resolución ARP dentro de la red local (petición para averiguar la MAC de 192.168.0.1 desde 192.168.0.37), así como algún frame Ethernet de broadcast de tipo no identificado por la herramienta. A continuación aparecen múltiples paquetes IP entre la máquina local 192.168.0.200 y distintos servidores externos en el puerto 443, tanto en UDP (probablemente tráfico QUIC) como en TCP, donde pueden verse flags de *push* y *finish* que indican el envío de datos de aplicaciones web y el cierre ordenado de las conexiones TLS.

```
> sudo tcpdump -i enp27s0 -n -c 20
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on enp27s0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
13:38:24.880293 ARP, Request who-has 192.168.0.1 tell 192.168.0.37, length 46
13:38:25.780269 00:5f:07:30:f3:70 > ff:ff:ff:ff:ff:ff, ethertype Unknown (0x8f83), length 60:
  0x0000: 0802 0002 0000 33ae 0000 0000 0003 7374 ..3....st
  0x0010: 0000 6439 0000 0000 0000 0000 0000 0000 ..d9.....
  0x0020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
13:38:26.436881 IP 192.168.0.200.56712 > 142.250.200.110.443: UDP, length 1250
13:38:26.484666 IP 142.250.200.110.443 > 192.168.0.200.56712: UDP, length 1250
13:38:26.485128 IP 192.168.0.200.56712 > 142.250.200.110.443: UDP, length 774
13:38:26.503207 IP 142.250.200.110.443 > 192.168.0.200.56712: UDP, length 24
13:38:26.510503 IP 192.168.0.200.56712 > 142.250.200.110.443: UDP, length 32
13:38:26.542296 IP 142.250.200.110.443 > 192.168.0.200.56712: UDP, length 28
13:38:26.553710 IP 142.250.200.110.443 > 192.168.0.200.56712: UDP, length 66
13:38:26.553935 IP 192.168.0.200.56712 > 142.250.200.110.443: UDP, length 35
13:38:26.618406 IP 142.250.200.110.443 > 192.168.0.200.56712: UDP, length 24
13:38:26.918768 IP 192.168.0.200.53016 > 141.95.47.140.443: Flags [P.], seq 2107674895:2107674919, ack 3016217431, win 615, options [nop,nop,TS val 1204892893 ecr 1447076448], length 24
13:38:26.918794 IP 192.168.0.200.53016 > 141.95.47.140.443: Flags [F.], seq 24, ack 1, win 615, option s [nop,nop,TS val 1204892893 ecr 1447076448], length 0
13:38:26.931665 IP 192.168.0.200.53022 > 141.95.47.140.443: Flags [P.], seq 3921099354:3921099378, ack 163463223, win 665, options [nop,nop,TS val 1204892905 ecr 1447076788], length 24
13:38:26.931696 IP 192.168.0.200.53022 > 141.95.47.140.443: Flags [F.], seq 24, ack 1, win 665, option s [nop,nop,TS val 1204892905 ecr 1447076788], length 0
13:38:26.970649 IP 141.95.47.140.443 > 192.168.0.200.53016: Flags [F.], seq 1, ack 24, win 504, options [nop,nop,TS val 1447085984 ecr 1204892893], length 0
13:38:26.970691 IP 192.168.0.200.53016 > 141.95.47.140.443: Flags [.], ack 2, win 615, options [nop,no p,TS val 1204892944 ecr 1447085984], length 0
13:38:26.974291 IP 141.95.47.140.443 > 192.168.0.200.53016: Flags [.], ack 25, win 504, options [nop,no op,TS val 1447085985 ecr 1204892893], length 0
13:38:26.983716 IP 141.95.47.140.443 > 192.168.0.200.53022: Flags [F.], seq 1, ack 25, win 504, options [nop,nop,TS val 1447085997 ecr 1204892905], length 0
13:38:26.983754 IP 192.168.0.200.53022 > 141.95.47.140.443: Flags [.], ack 2, win 665, options [nop,no p,TS val 1204892958 ecr 1447085997], length 0
20 packets captured
20 packets received by filter
0 packets dropped by kernel
```

Figura 12.1: Salida del comando `tcpdump`