

# Práctica 1: Entorno de trabajo auditoría de redes inalámbricas

Seguridad de las Comunicaciones Inalámbricas

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

**Fecha:** 7 de diciembre de 2025

# Índice general

<b>1. Análisis y Configuración de Interfaces de Red</b>	<b>1</b>
1.1. Enumeración de Interfaces . . . . .	1
<b>2. Gestión del Estado de la Interfaz y Escalado de Privilegios</b>	<b>2</b>
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
<b>3. Verificación de la Pila TCP/IP y Conectividad Externa</b>	<b>5</b>
3.1. Prueba de Loopback . . . . .	5
3.2. Prueba de Conectividad Externa . . . . .	6
<b>4. Análisis de la Caché del Protocolo ARP</b>	<b>6</b>
<b>5. Inspección de la Tabla de Enrutamiento</b>	<b>7</b>
5.1. Análisis . . . . .	7
<b>6. Trazado de Ruta de Red (Traceroute)</b>	<b>7</b>
6.1. Objetivo . . . . .	7
6.2. Desarrollo . . . . .	7
6.3. Resultados . . . . .	7
<b>7. Escaneo de Puertos Locales con Nmap</b>	<b>8</b>
7.1. Objetivo . . . . .	8
7.2. Desarrollo . . . . .	8
7.3. Análisis . . . . .	8
<b>8. Análisis de Sockets y Conexiones de Red</b>	<b>8</b>
8.1. Objetivo . . . . .	8
8.2. Desarrollo . . . . .	8
8.2.1. Comando tradicional . . . . .	8
8.2.2. Herramienta moderna . . . . .	8
8.3. Análisis . . . . .	8
<b>9. Resolución de Nombres DNS (NSLookup)</b>	<b>8</b>
9.1. Objetivo . . . . .	8
9.2. Desarrollo . . . . .	8
9.3. Documentación . . . . .	9
<b>10. Interacción de Red con Netcat (nc)</b>	<b>9</b>
10.1. Objetivo . . . . .	9
10.2. Desarrollo . . . . .	9
10.2.1. Revisión de opciones . . . . .	9

10.2.2. Sintaxis clave . . . . .	9
<b>11.Reconocimiento DNS Avanzado con dnsenum</b>	<b>9</b>
11.1. Objetivo . . . . .	9
11.2. Desarrollo . . . . .	9
11.3. Análisis . . . . .	9
<b>12.Análisis de Tráfico en Terminal con tcpdump</b>	<b>10</b>
12.1. Objetivo . . . . .	10
12.2. Desarrollo . . . . .	10
12.2.1. Explicación de flags . . . . .	10
12.3. Observaciones . . . . .	10

# 1. Análisis y Configuración de Interfaces de Red

Ejecutar el comando para listar todas las interfaces de red:

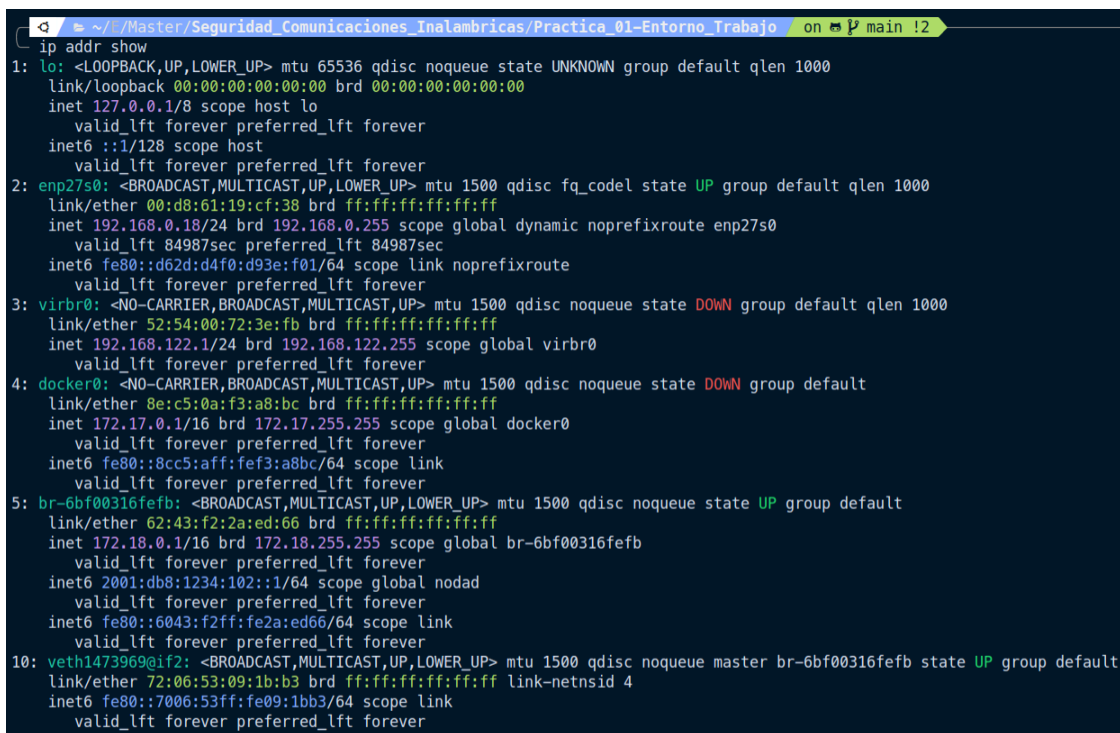
```
1 ip addr show
```

Forma abreviada: `ip a`

## 1.1. Enumeración de Interfaces

Para cada interfaz activa (eth0, lo, wlan0), documentar:

- **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:f01/64 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::8cc5:aff:fef3:a8bc/64 scope link
        valid_lft forever preferred_lft forever
5: br-6bf00316febf: <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-6bf00316febf
        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-6bf00316febf 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`

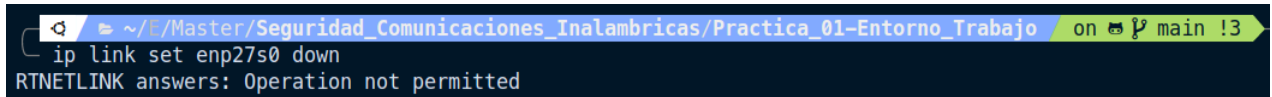
## 2. Gestión del Estado de la Interfaz y Escalado de Privilegios

### 2.1. Intento sin privilegios

Intentar deshabilitar la interfaz eth0:

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

*Resultado esperado:* Error de permiso denegado.



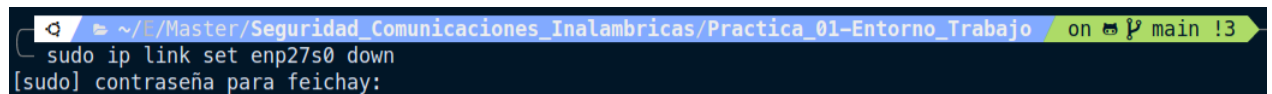
```
~ /c/Master/Seguridad_Comunicaciones_Inalambricas/Practica_01-Entorno_Trabajo on main !3  
ip link set enp27s0 down  
RTNETLINK answers: Operation not permitted
```

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
```



```
~ /c/Master/Seguridad_Comunicaciones_Inalambricas/Practica_01-Entorno_Trabajo on main !3  
sudo ip link set enp27s0 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
```

```
~ /Master/Seguridad Comunicaciones Inalambricas/Practica 01-Entorno Trabajo on main !5
sudo ip link set enp27s0 down

~ /Master/Seguridad Comunicaciones Inalambricas/Practica 01-Entorno Trabajo on main !5
ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
   link/loopback 00:00:00:00:00:00 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> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
   link/ether 00:08:01:19: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 86307sec preferred_lft 86307sec
   inet6 fe80::94ee:b900:39f7:9349/64 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 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::8cc5:aff:fe3:a8bc/64 scope link
       valid_lft forever preferred_lft forever
5: br-6bf00316febf: <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-6bf00316febf
       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: veth1473969696i2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-6bf00316febf 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 2.3: Verificación de estado inactivo de eth0

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

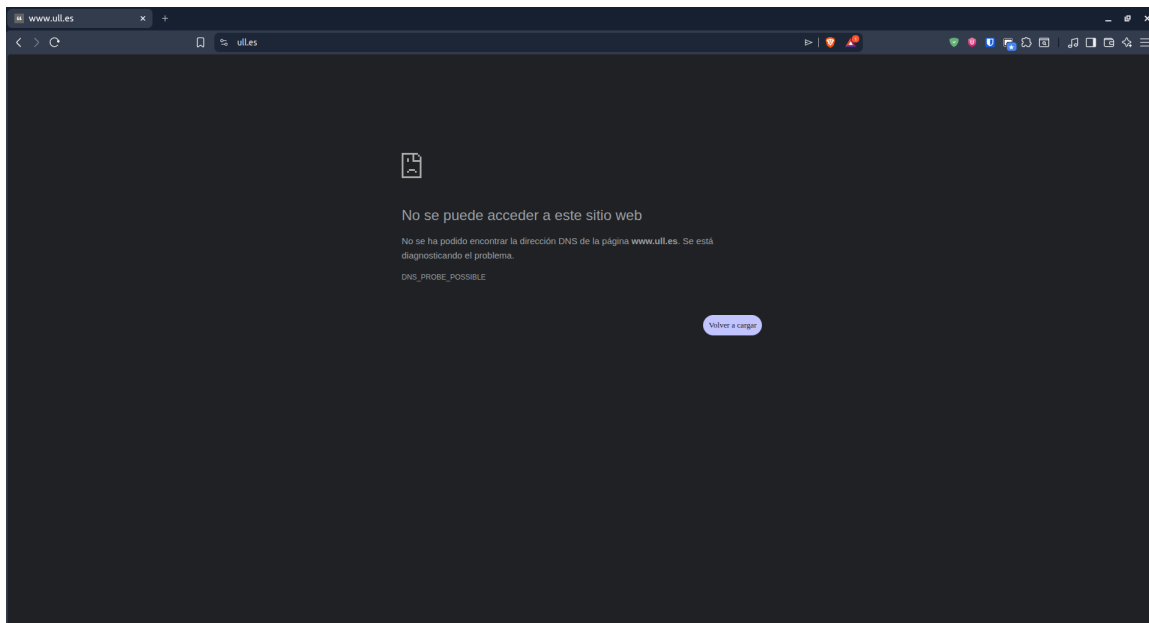


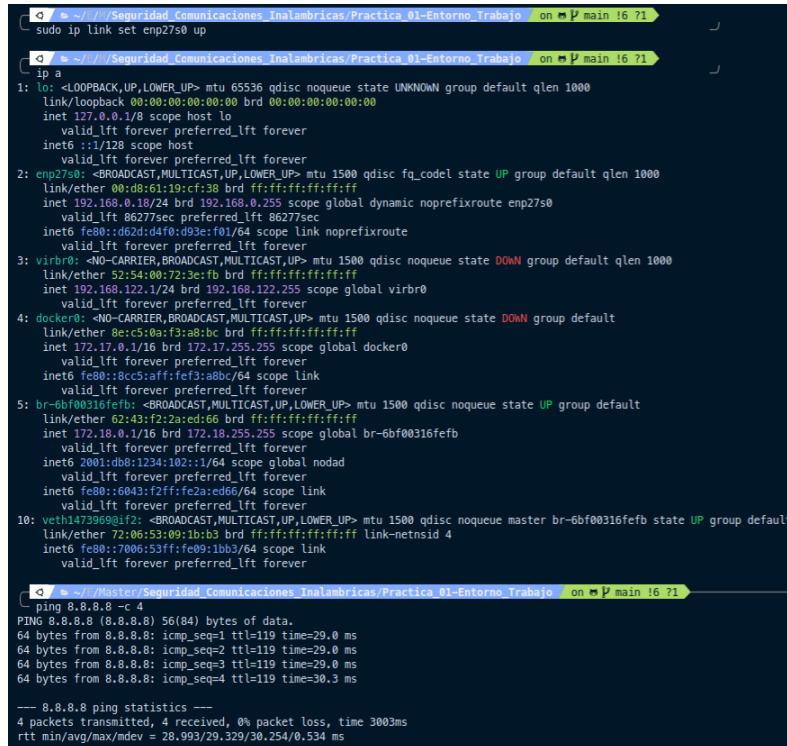
Figura 2.4: Pérdida de conexión

## 2.4. Reactivación

Restaurar la interfaz:

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

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



```
sudo ip link set enp27s0 up

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: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 86277sec preferred_lft 86277sec
    inet6 fe80::d62d:d4f0:d93e:f01/64 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 9e:63:0a: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::8cc5:aff:fe3:a8bc/64 scope link
        valid_lft forever preferred_lft forever
5: br-6bf00316feb: <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-6bf00316feb
        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: veth147396981f2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-6bf00316feb 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

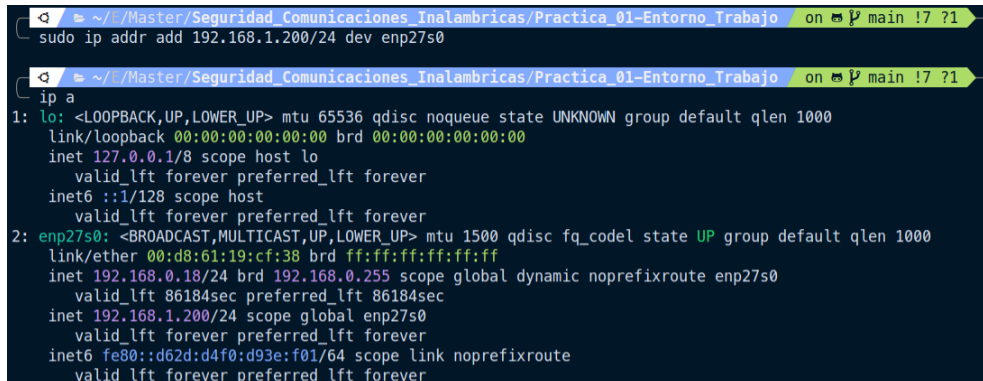
ping 8.8.8.8 -c 4
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data:
64 bytes from 8.8.8.8: icmp_seq=1 ttl=119 time=29.0 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=119 time=29.0 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=119 time=29.0 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=119 time=30.3 ms

--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
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
```

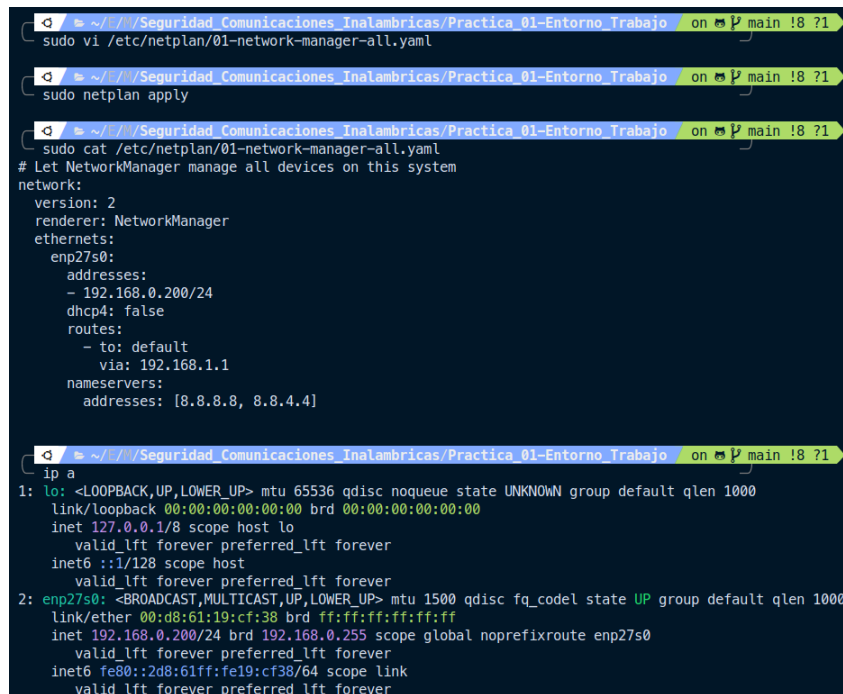


```
sudo ip addr add 192.168.1.200/24 dev enp27s0

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: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 86184sec preferred_lft 86184sec
    inet 192.168.1.200/24 scope global enp27s0
        valid_lft forever preferred_lft forever
    inet6 fe80::d62d:d4f0:d93e:f01/64 scope link noprefixroute
        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/network/interfaces`.



```
~$ sudo vi /etc/netplan/01-network-manager-all.yaml
~$ sudo netplan apply
~$ sudo cat /etc/netplan/01-network-manager-all.yaml
# 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:d8:61:19:cf: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::2d8:61ff:fe19:cf38/64 scope link
       valid_lft forever preferred_lft forever
```

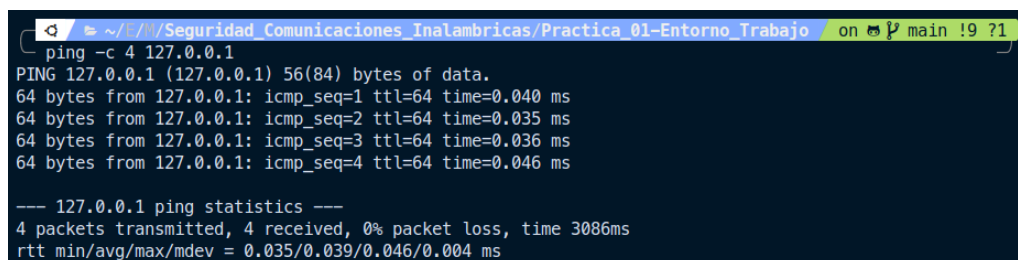
Figura 2.7: Verificación de nueva IP asignada a eth0

### 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
```

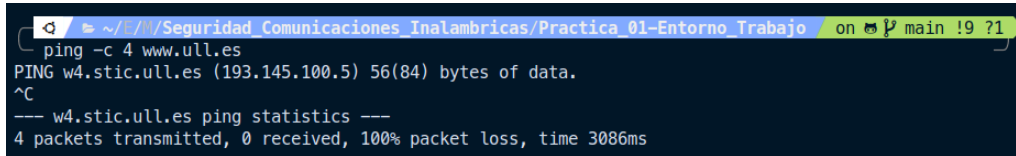
Figura 3.1: Prueba de conectividad a loopback



## 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.

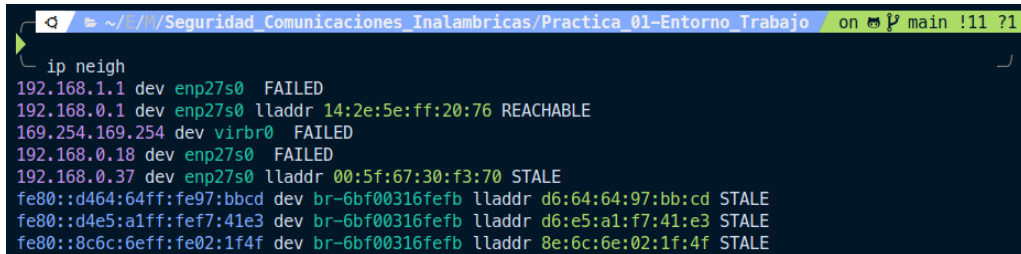
## 4. Análisis de la Caché del Protocolo ARP

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-6bf00316feb lladdr d6:64:64:97:bb:cd STALE
fe80::d4e5:a1ff:fef7:41e3 dev br-6bf00316feb lladdr d6:e5:a1:f7:41:e3 STALE
fe80::8c6c:6eff:fe02:1f4f dev br-6bf00316feb 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`

### 5.1. Análisis

Identificar la ruta por defecto (`default via ...`) que indica la IP de la puerta de enlace.

## 6. Trazado de Ruta de Red (Traceroute)

### 6.1. Objetivo

Determinar los saltos (routers intermedios) entre la máquina y un destino.

### 6.2. Desarrollo

Realizar trazados de ruta:

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

### 6.3. Resultados

Documentar:

- Número total de saltos para cada destino
- Latencia en cada nodo

## 7. Escaneo de Puertos Locales con Nmap

### 7.1. Objetivo

Identificar puertos que están escuchando y servicios asociados en la máquina local.

### 7.2. Desarrollo

Escaneo con detección de versión:

```
1 nmap -sV localhost
```

El flag `-sV` intenta determinar la versión del servicio.

### 7.3. Análisis

Esto simula un reconocimiento inicial desde la perspectiva de un atacante en la misma red.

## 8. Análisis de Sockets y Conexiones de Red

### 8.1. Objetivo

Enumerar conexiones activas y puertos abiertos.

### 8.2. Desarrollo

#### 8.2.1. Comando tradicional

```
1 netstat -tulpn
```

#### 8.2.2. Herramienta moderna

Usando `ss` (socket statistics):

```
1 ss -tulpn
```

### 8.3. Análisis

Buscar puertos abiertos inesperados o conexiones remotas `ESTABLISHED` sospechosas.

## 9. Resolución de Nombres DNS (NSLookup)

### 9.1. Objetivo

Realizar consultas DNS para resolver nombres de dominio.

### 9.2. Desarrollo

Consultas con `nslookup`:

```
1 nslookup www.u11.es
2 nslookup www.w3c.org
```

## 9.3. Documentación

Identificar y documentar:

- Servidor DNS que resuelve la consulta (indicado como **Server**)
- Registros A (IPv4) resueltos para cada dominio

## 10. Interacción de Red con Netcat (nc)

### 10.1. Objetivo

Comprender el uso de Netcat para depuración y explotación de redes.

### 10.2. Desarrollo

#### 10.2.1. Revisión de opciones

```
1 nc -h
```

#### 10.2.2. Sintaxis clave

##### 1. Modo Escucha (Listener):

```
1 nc -l -p 1234
2
```

##### 2. Modo Cliente:

```
1 nc <IP_remota> <puerto>
2
```

## 11. Reconocimiento DNS Avanzado con dnsenum

### 11.1. Objetivo

Recopilar inteligencia de fuentes abiertas (OSINT) sobre un dominio.

### 11.2. Desarrollo

Ejecutar **dnsenum**:

```
1 dnsenum tecnomobile.com
```

### 11.3. Análisis

Extraer información clave:

1. Registros de Host (A/AAAA)
2. Servidores de Nombres (NS)
3. Servidores de Correo (MX)
4. Sub-enumeración de subdominios

## 12. Análisis de Tráfico en Terminal con tcpdump

### 12.1. Objetivo

Analizar paquetes de red en línea de comandos.

### 12.2. Desarrollo

Captura básica (requiere privilegios):

```
1 sudo tcpdump -i eth0 -n -c 20
```

#### 12.2.1. Explicación de flags

- `-i eth0`: Escucha en la interfaz eth0
- `-n`: No resuelve nombres DNS/IPs
- `-c 20`: Captura 20 paquetes y se detiene

### 12.3. Observaciones

Observar el flujo de tráfico en tiempo real y documentar los datos relevantes.