

# EJERCICIOS DE RED LINUX

1. Muestra todas las interfaces de red activas y sus direcciones IP en el sistema.

```
$ ip addr show
```

```
sergio@sergio:~/multimedia_pruebas/video$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    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 noprefixroute
            valid_lft forever preferred_lft forever
```

2. ¿Cómo mostrarías solo la información de la interfaz de red `eth0` usando `ip a`?

```
$ ip a show eth0
```

```
sergio@sergio:~$ sudo ip link set eth0 up
sergio@sergio:~$ ip a show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP
    link/ether 08:00:27:2d:92:76 brd ff:ff:ff:ff:ff:ff
        queueingdiscipline: fq_codel
```

3. Configura manualmente la dirección IP `192.168.1.100/24` en la interfaz `eth0` con `ifconfig`.

```
$ sudo ifconfig eth0 192.168.1.100 netmask 255.255.255.0
```

4. Envía 10 paquetes ICMP a la dirección IP `8.8.8.8` usando `ping`.

```
$ ping -c 10 8.8.8.8
```

```
sergio@sergio:~$ ping -c 10 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
From 192.168.1.100 icmp_seq=1 Destination Host Unreachable
From 192.168.1.100 icmp_seq=2 Destination Host Unreachable
From 192.168.1.100 icmp_seq=3 Destination Host Unreachable
From 192.168.1.100 icmp_seq=4 Destination Host Unreachable
```

##### 5. Consulta la dirección IP de `www.example.com` usando `nslookup`

```
$ nslookup www.example.com
```

```
sergio@sergio:~$ nslookup www.example.com
;; Got SERVFAIL reply from 127.0.0.53
Server:      127.0.0.53
Address:     127.0.0.53#53

** server can't find www.example.com: SERVFAIL
```

##### 8. Consulta el nombre del host actual del sistema.

```
$ hostname
```

```
sergio@sergio:~$ hostname
sergio
```

##### 9. Obtén la información de registro del dominio `example.com` usando `whois`.

```
$ whois example.com
```

##### 10. Cambia temporalmente el nombre del host a `servidor01` usando `hostname`

```
$ sudo hostname servidor01
$ hostname
```

```
sergio@sergio:~$ sudo hostname servidor01
sergio@sergio:~$ who
sergio    seat0          2025-11-04 17:43 (login screen)
sergio    tty2          2025-11-04 17:43 (tty2)
sergio@sergio:~$ hostname
servidor01
```

12. Muestra las estadísticas de la red, como la cantidad de paquetes transmitidos, usando `netstat`

```
$ ping 192.168.1.1
```

```
sergio@sergio:~$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
From 192.168.1.100 icmp_seq=1 Destination Host Unreachable
From 192.168.1.100 icmp_seq=2 Destination Host Unreachable
From 192.168.1.100 icmp_seq=3 Destination Host Unreachable
From 192.168.1.100 icmp_seq=4 Destination Host Unreachable
```

13. \*Realiza una consulta inversa para obtener el nombre de dominio asociado a la IP `8.8.8.8` con `nslookup`

```
$ nslookup 8.8.8.8
```

```
nslookup: no se encontró la orden
sergio@sergio:~$ nslookup 8.8.8.8
8.8.8.8.in-addr.arpa      name = dns.google.
```

14. Configura temporalmente la máscara de subred `255.255.255.128` en la interfaz `eth1` usando `ifconfig`.

```
$ sudo ifconfig eth1 192.168.1.10 netmask 255.255.255.128 up
```

15. Muestra las rutas de enrutamiento actuales usando `netstat`.

```
$ netstat -r
```

```
sergio@sergio:~$ netstat -r
Tabla de rutas IP del núcleo
Destino      Pasarela      Genmask      Indic      MSS Ventan
z
default      _gateway      0.0.0.0      UG          0 0
10.142.60.0  0.0.0.0      255.255.252.0  U          0 0
sergio@sergio:~$
```

16. \*Realiza una solicitud HTTP GET a la API de GitHub para obtener los repositorios de `usuario123` usando `curl`.