

# Comandos esenciales de red en Linux para ciberseguridad

## Diagnóstico de red básico

### ping

Verifica conectividad con otra IP o dominio.


 Ejemplo:

ping 8.8.8.8

```
~ > ping 8.8.8.8
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=255 time=14.5 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=255 time=14.5 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 14.506/14.507/14.508/0.001 ms
```

### ip

Sustituye a ifconfig. Gestiona interfaces, direcciones y rutas.

 Ejemplos:

ip addr show # Ver direcciones IP de interfaces

```
~ > 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 noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP qlen 1000
    link/ether 08:00:27:00:00:00 brd ff:ff:ff:ff:ff:ff
```

ip link set eth0 up # Activar interfaz eth0

```
~ > sudo ip link set enp0s3 up
[sudo] password for osboxes:
```

ip route show # Ver rutas de red configuradas

```
~ > ip route show
default via 10.0.2.2 dev enp0s3 proto dhcp src 10.0.2.15 metric 100
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.15 metric 100
```

### traceroute

Muestra la ruta que siguen los paquetes hasta un destino.

■ Ejemplo:

tracert google.com

En sistemas donde no está instalado:

sudo apt install traceroute

```
~ > traceroute google.es
traceroute to google.es (142.250.200.131), 30 hops max, 60 byte packets
 1  gateway (10.0.2.2)  1.981 ms  1.997 ms  1.989 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  *^C
```

**netstat (reemplazado por ss, pero aún útil)**

**Muestra conexiones activas, puertos y rutas.**

■ Ejemplos:

Hay que instalar esto:

netstat -tuln # Puertos escuchando (TCP/UDP, sin nombres)

```
~ > netstat -tuln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.54:53           0.0.0.0:*               LISTEN
tcp6       0      0 :::1:631                :::*                    LISTEN
udp        0      0 0.0.0.0:34731           0.0.0.0:*
udp        0      0 10.0.2.15:3702          0.0.0.0:*
udp        0      0 239.255.255.250:3702    0.0.0.0:*
udp        0      0 127.0.0.54:53           0.0.0.0:*
udp        0      0 127.0.0.53:53           0.0.0.0:*
udp        0      0 0.0.0.0:5353            0.0.0.0:*
```

netstat -r # Tabla de rutas

```
~ > netstat -r
Kernel IP routing table
Destination Gateway Genmask Flags MSS Window irtt Iface
default gateway 0.0.0.0 UG 0 0 0 enp0s3
10.0.2.0 0.0.0.0 255.255.255.0 U 0 0 0 enp0s3
```

Para sistemas sin netstat:

sudo apt install net-tools

---

**nmap**

## Escáner de puertos y servicios en red.

 Ejemplo:

`nmap -sV 192.168.1.1`

Muestra qué servicios hay en ejecución y sus versiones.

```
~ > nmap -sV 10.0.2.15
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-19 03:35 EDT
Nmap scan report for osboxes (10.0.2.15)
Host is up (0.000070s latency).
All 1000 scanned ports on osboxes (10.0.2.15) are in ignored states.
Not shown: 1000 closed tcp ports (conn-refused)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.25 seconds
```

## lsof

Muestra archivos abiertos, incluyendo sockets de red.

 Ejemplo:

`sudo lsof -i`

Muestra:

- Puertos abiertos
- Conexiones activas
- Servicios escuchando

```
~ > sudo lsof -i
COMMAND  PID    USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
systemd-r 298  systemd-resolve 14u  IPv4  4333      0t0  UDP  _localdnssstub:domain
systemd-r 298  systemd-resolve 15u  IPv4  4334      0t0  TCP  _localdnssstub:domain (LISTEN)
systemd-r 298  systemd-resolve 16u  IPv4  4335      0t0  UDP  _localdnssproxy:domain
systemd-r 298  systemd-resolve 17u  IPv4  4336      0t0  TCP  _localdnssproxy:domain (LISTEN)
avahi-dae 988    avahi   12u  IPv4  10923     0t0  UDP  *:mdns
avahi-dae 988    avahi   13u  IPv6  10924     0t0  UDP  *:mdns
avahi-dae 988    avahi   14u  IPv4  10925     0t0  UDP  *:34731
avahi-dae 988    avahi   15u  IPv6  10926     0t0  UDP  *:56898
NetworkMa 1082   root    26u  IPv4  12190     0t0  UDP  osboxes:bootpc->_gateway:bootps
cupsd     1262   root     6u  IPv6  12082     0t0  TCP  ip6-localhost:ipp (LISTEN)
cupsd     1262   root     7u  IPv4  12083     0t0  TCP  localhost:ipp (LISTEN)
python3   2745   osboxes  7u  IPv4  19421     0t0  UDP  239.255.255.250:3702
python3   2745   osboxes  8u  IPv4  19422     0t0  UDP  *:34056
python3   2745   osboxes  9u  IPv4  19423     0t0  UDP  osboxes:3702
python3   2745   osboxes 10u  IPv4  19426     0t0  UDP  [ff02::c]:3702
python3   2745   osboxes 11u  IPv6  19427     0t0  UDP  *:43360
python3   2745   osboxes 12u  IPv6  19428     0t0  UDP  osboxes:3702
firefox   2831   osboxes 130u  IPv4  20726     0t0  TCP  osboxes:46530->93.243.107.34.bc.go
```

## Monitorización avanzada en tiempo real

### iftop

Monitoriza tráfico en tiempo real por interfaz.

- ◆ Muestra conexiones IP → IP, con tasas de transferencia.

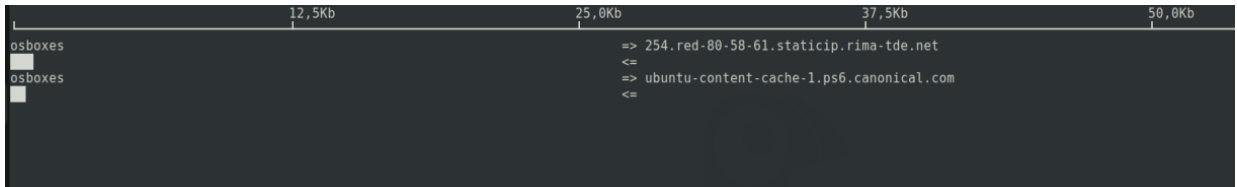
## Instalación:

```
sudo apt update
```

```
sudo apt install iftop
```

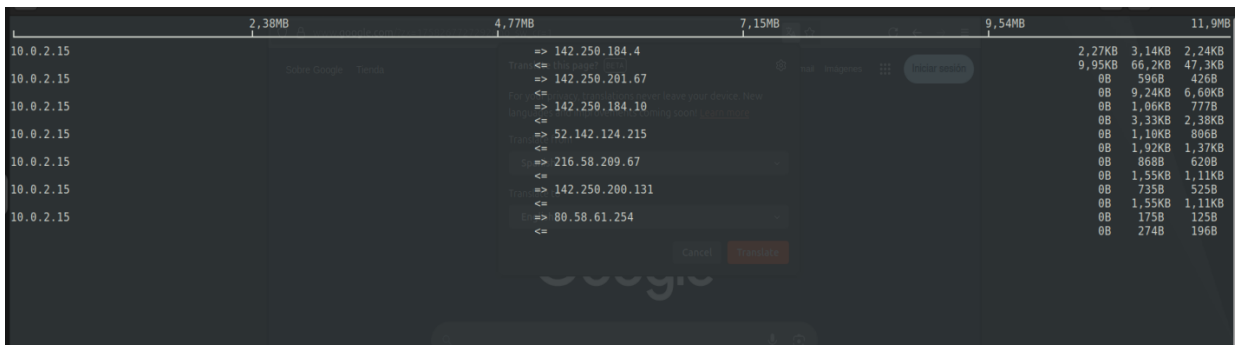
## Uso básico:

```
sudo iftop
```



## Ejemplo completo:

```
sudo iftop -i eth0 -n -N -B
```



## Opción Función

- i Selecciona interfaz (eth0, wlan0...)
- n No resuelve nombres de host
- N No resuelve puertos
- B Usa bytes en vez de bits
- P Muestra puertos en la vista

## Atajos de teclado útiles en iftop:

### Tecla Acción

- h Ayuda
- S Ordenar por salida
- R Ordenar por entrada

## Tecla Acción

B Bits/bytes


q Salir

### vnstat

#### Control de tráfico de red con histórico (diario, mensual, etc.).

 Instalación:

```
sudo apt install vnstat
```

 Uso en tiempo real:

```
vnstat -l
```

```
~ > vnstat -l
Monitoring enp0s3... (press CTRL-C to stop)

/      rx:          0 bit/s    0 p/s      tx:          0 bit/s    0 p/s^C

enp0s3 / traffic statistics

              rx          |          tx
-----+-----
bytes          532,47 KiB |          53,83 KiB
-----+-----
      max          1,82 Mbit/s |          136,22 kbit/s
    average          103,86 kbit/s |          10,50 kbit/s
      min              0 bit/s |              0 bit/s
-----+-----
packets           514 |           211
-----+-----
      max           194 p/s |          74 p/s
    average           12 p/s |          5 p/s
      min              0 p/s |              0 p/s
-----+-----
time              42 seconds

~ > vnstat -l
Monitoring enp0s3... (press CTRL-C to stop)

/      rx:          0 bit/s    0 p/s      tx:          0 bit/s    0 p/s
```

 Consultas históricas:

```
vnstat # Resumen por días y meses
```

```
vnstat -d # Por días
```

```
vnstat -m # Por meses
```

```

~ > vnstat
enp0s3: No data. Timestamp of last update is same 2025-09-19 03:44:16 as of database creation.
~ > vnstat -d
enp0s3: No data. Timestamp of last update is same 2025-09-19 03:44:16 as of database creation.
~ > vnstat -m
enp0s3: No data. Timestamp of last update is same 2025-09-19 03:44:16 as of database creation.


```

## iptraf

### Monitor gráfico en terminal para tráfico IP detallado.

 Instalación:

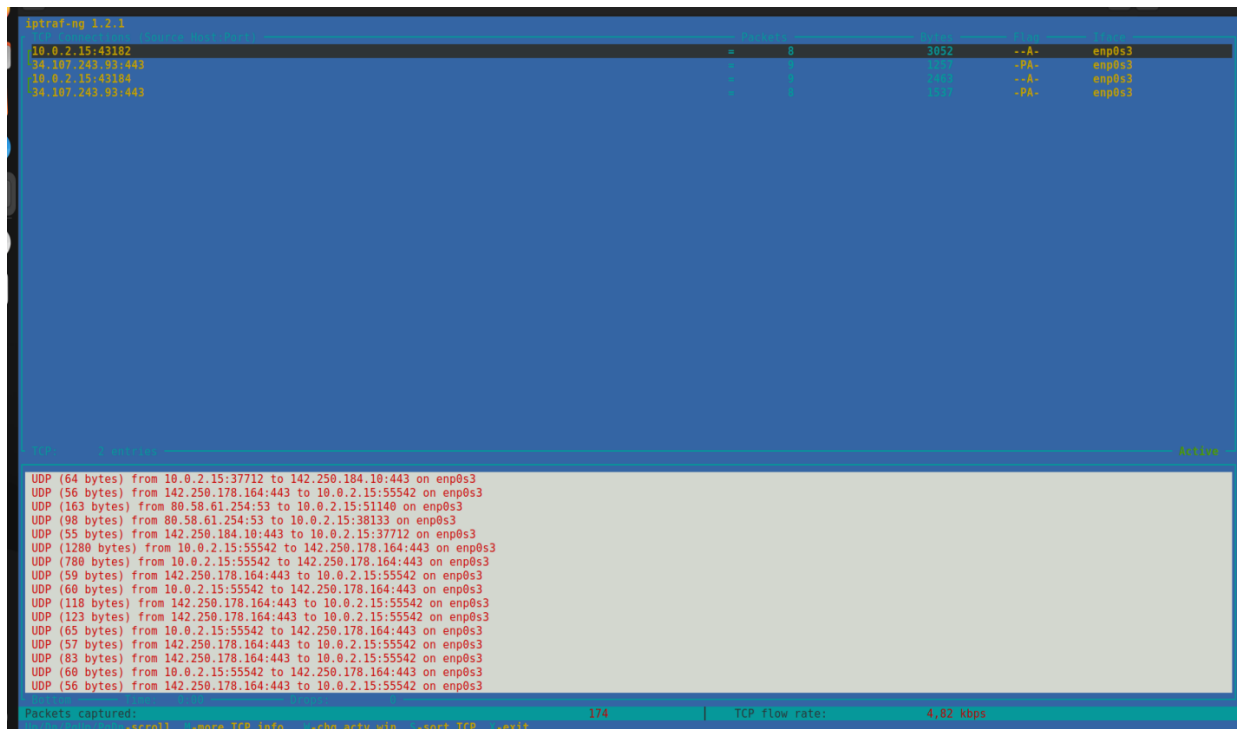
`sudo apt install iptraf`

 Ejecución:


`sudo iptraf`

◆ Muestra:

- Estadísticas por IP
- Tráfico entrante/saliente
- Estadísticas de caída
- Información LAN útil para auditorías



### Ejemplo de actividad práctica para clase

 **Objetivo:** detectar qué equipo en la red está generando más tráfico

1. Ejecuta `sudo iftop -i eth0`



```
sudo iptraf-i enp0s3

iptraf-ng 1.2.1
TCP connections (Source Host:Port)
10.0.2.15:43182
34.107.243.93:443

TCP: 1 entries

UDP (59 bytes) from 216.58.215.170:443 to 10.0.2.15:45850 on enp0s3
UDP (377 bytes) from 216.58.215.170:443 to 10.0.2.15:45850 on enp0s3
UDP (261 bytes) from 216.58.215.170:443 to 10.0.2.15:45850 on enp0s3
UDP (66 bytes) from 10.0.2.15:45850 to 216.58.215.170:443 on enp0s3
UDP (55 bytes) from 216.58.215.170:443 to 10.0.2.15:45850 on enp0s3
UDP (1280 bytes) from 10.0.2.15:37563 to 142.250.178.164:443 on enp0s3
UDP (779 bytes) from 10.0.2.15:37563 to 142.250.178.164:443 on enp0s3
UDP (59 bytes) from 142.250.178.164:443 to 10.0.2.15:37563 on enp0s3
UDP (60 bytes) from 10.0.2.15:37563 to 142.250.178.164:443 on enp0s3
UDP (117 bytes) from 142.250.178.164:443 to 10.0.2.15:37563 on enp0s3
```

- 2.
3. Observa las IPs con más transferencia
4. Usa ping o nmap para ver si están activas
5. Opcional: revisar la tabla ARP con arp -a

```
~ > arp -a
gateway (10.0.2.2) at 52:55:0a:00:02:02 [ether] on enp0s3
~ >
```

6.

## ps aux (Linux)

Lista los procesos en ejecución.

 Ejemplo:

ps aux

```

~ > ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.1  0.2 25304 15976 ?        Ss   03:25   0:01 /sbin/init splash
root         2  0.0  0.0      0      0 ?        S    03:25   0:00 [kthreadd]
root         3  0.0  0.0      0      0 ?        S    03:25   0:00 [pool_workqueue_release]
root         4  0.0  0.0      0      0 ?        I<   03:25   0:00 [kworker/R-rcu_gp]
root         5  0.0  0.0      0      0 ?        I<   03:25   0:00 [kworker/R-sync_wq]
root         6  0.0  0.0      0      0 ?        I<   03:25   0:00 [kworker/R-kvfree_rcu_reclaim]
root         7  0.0  0.0      0      0 ?        I<   03:25   0:00 [kworker/R-slub_flushwq]
root         8  0.0  0.0      0      0 ?        I<   03:25   0:00 [kworker/R-netns]
root        11  0.0  0.0      0      0 ?        I<   03:25   0:00 [kworker/0:0H-events_highpri]
root        12  0.1  0.0      0      0 ?        I    03:25   0:02 [kworker/u4:0-events_power_efficient]
root        13  0.0  0.0      0      0 ?        I<   03:25   0:00 [kworker/R-mm_percpu_wq]
root        14  0.0  0.0      0      0 ?        I    03:25   0:00 [rcu_tasks_kthread]
root        15  0.0  0.0      0      0 ?        I    03:25   0:00 [rcu_tasks_rude_kthread]
root        16  0.0  0.0      0      0 ?        I    03:25   0:00 [rcu_tasks_trace_kthread]
root        17  0.0  0.0      0      0 ?        S    03:25   0:00 [ksoftirqd/0]
root        18  0.0  0.0      0      0 ?        I    03:25   0:00 [rcu_preempt]
root        19  0.0  0.0      0      0 ?        S    03:25   0:00 [rcu_exp_par_gp_kthread_worker/0]
root        20  0.0  0.0      0      0 ?        S    03:25   0:00 [rcu_exp_gp_kthread_worker]
root        21  0.0  0.0      0      0 ?        S    03:25   0:00 [migration/0]

```

💡 Útil para detectar procesos maliciosos.