Linux_

Guía de atajos · comandos, redes y file system

```
user@dajomar:~$ man linux
```

Comandos Básicos

Ayuda de un comando

```
man ls     # manual completo
ls --help  # ayuda rápida
tldr tar  # ejemplos prácticos (si está instalado)
```

¿Dónde estoy y qué hay aquí?

```
pwd
ls -lah  # lista larga, ocultos, tamaños legibles
```

`-l` formato largo · `-a` incluye ocultos · `-h` tamaños legibles (K, M, G)

Moverse entre directorios

```
cd /var/log  # ruta absoluta
cd ..        # subir un nivel
cd ~  ó  cd  # ir al home
cd -         # volver al directorio anterior
```

Crear, copiar, mover y borrar

```
mkdir -p proyectos/app/src
touch notas.txt
cp archivo.txt /tmp/
cp -r carpeta/ respaldo/  # recursivo
mv viejo.txt nuevo.txt    # mover o renombrar
rm archivo.txt
rm -r carpeta/            # ⚠ sin papelera
```

Buscar archivos

```
find / -name "*.conf" 2>/dev/null
find . -type f -size +100M
find /var/log -mtime -1  # modificados último día
which nginx              # ruta de un ejecutable
```

Pro Tip:
`Tab` autocompleta · `Ctrl+R` busca en el historial · `!!` repite el último comando · `sudo !!` lo repite como root · `Ctrl+C` cancela · `Ctrl+L` limpia pantalla.

Ver y procesar texto

```
cat archivo.txt
# q para salir, / para buscar
less /var/log/syslog
head -n 20 archivo.txt
tail -f /var/log/auth.log  # seguir el log en vivo
```

Filtrar y contar

```
grep -i "error" app.log
# recursivo + nº de línea
grep -rn "password" /etc/
wc -l archivo.txt
sort datos.txt | uniq -c | sort -nr
# 1ª columna separada por :
cut -d: -f1 /etc/passwd
awk '{print $1}' access.log
sed -i 's/viejo/nuevo/g' conf.txt
```

Pipes y redirecciones

`cmd1 | cmd2`	La salida de cmd1 entra a cmd2
`> / >>`	Escribe / agrega la salida a un archivo
`2> err.log`	Redirige solo los errores
`&> todo.log`	Salida y errores al mismo archivo
`cmd1 && cmd2`	cmd2 solo corre si cmd1 tuvo éxito
`cmd &`	Ejecuta en segundo plano

Permisos y propietarios

Leer `-rwxr-x---`: tipo · **usuario** · **grupo** · **otros**

Letra	Valor	Archivo / Directorio
r	4	leer / listar contenido
w	2	modificar / crear y borrar dentro
x	1	ejecutar / entrar (cd)

```
chmod 755 script.sh     # rwx r-x r-x
chmod 640 secreto.conf  # rw- r-- ---
chmod u+x script.sh     # +x solo al dueño
chown david:devs archivo.txt
chown -R www-data: /var/www
```

Usuarios y grupos

```
whoami ; id
sudo -i  # shell como root
sudo useradd -m -s /bin/bash ana
sudo passwd ana
# agregar a grupo (-a ¡importante!)
sudo usermod -aG sudo ana
groups ana
sudo userdel -r ana
```

Archivos clave: `/etc/passwd` usuarios · `/etc/shadow` hashes · `/etc/group` grupos · `/etc/sudoers` (editar con `visudo`).

Procesos y servicios

```
ps aux | grep nginx
top        ó   htop  # monitor en vivo
kill PID             # pide terminar (SIGTERM)
kill -9 PID          # fuerza (SIGKILL)
pkill -f nombre
```

systemd: administrar servicios

```
systemctl status ssh
sudo systemctl start|stop|restart ssh
# arranca y al boot
sudo systemctl enable --now nginx
systemctl list-units --type=service --state=failed
journalctl -u nginx -f  # logs del servicio
journalctl -p err -b    # errores desde el boot
```

Sistema y recursos

```
uname -a ; cat /etc/os-release
uptime  # tiempo encendido y carga
free -h                 # memoria RAM / swap
df -h                   # espacio por partición
du -sh /var/* | sort -h  # qué ocupa espacio
lscpu ; lsusb ; lspci
```

Paquetes

Acción	Debian / Ubuntu	RHEL / Fedora
Actualizar	`apt update && apt upgrade`	`dnf upgrade`
Instalar	`apt install htop`	`dnf install htop`
Eliminar	`apt remove htop`	`dnf remove htop`
Buscar	`apt search nginx`	`dnf search nginx`

Anteponer `sudo` a todos.

Compresión

```
tar -czvf respaldo.tar.gz carpeta/  # crear
tar -xzvf respaldo.tar.gz           # extraer
tar -tzvf respaldo.tar.gz  # ver contenido
zip -r app.zip app/ ; unzip app.zip
```

`-c` crear · `-x` extraer · `-z` gzip · `-v` verbose · `-f` archivo

Administración de redes

Interfaces, IPs y rutas

```
ip a  # interfaces y direcciones IP
ip link set eth0 up|down
ip r  # tabla de rutas / gateway
# temporal
sudo ip addr add 192.168.1.50/24 dev eth0
hostnamectl set-hostname srv01
```

Conectividad y diagnóstico

```
ping -c 4 8.8.8.8
traceroute google.com    ó   mtr google.com
dig google.com +short ; nslookup google.com
curl -I https://dajomar.com  # solo cabeceras HTTP
wget https://sitio.com/archivo.iso
```

Puertos y conexiones

```
ss -tulpn           # puertos en escucha + proceso
ss -tan state established
sudo lsof -i :443   # quién usa el puerto 443
nc -zv 10.0.0.5 22  # ¿puerto abierto?
# escaneo (solo en tu red)
sudo nmap -sV 192.168.1.0/24
```

`-t` TCP · `-u` UDP · `-l` escuchando · `-p` proceso · `-n` sin resolver nombres

Configuración persistente

```
nmcli device status  # NetworkManager
nmcli con show
sudo nmcli con mod "Wired 1" \
  ipv4.addresses 192.168.1.50/24 \
  ipv4.gateway 192.168.1.1 \
  ipv4.dns 1.1.1.1 ipv4.method manual
sudo netplan apply   # Ubuntu: /etc/netplan/*.yaml
```

`/etc/hosts` resolución local · `/etc/resolv.conf` servidores DNS · `/etc/services` puertos conocidos.

Firewall

```
sudo ufw status verbose       # Ubuntu
sudo ufw allow 22/tcp ; sudo ufw enable
sudo firewall-cmd --list-all  # RHEL / Fedora
sudo firewall-cmd --add-service=http --permanent
sudo nft list ruleset  # reglas de bajo nivel
```

Acceso remoto y transferencia

```
ssh usuario@10.0.0.5 -p 22
ssh-keygen -t ed25519
# login sin contraseña
ssh-copy-id usuario@10.0.0.5
scp archivo.txt usuario@host:/tmp/
rsync -avz carpeta/ usuario@host:/backup/
```

Captura de tráfico

```
sudo tcpdump -i eth0 port 53
sudo tcpdump -i any host 10.0.0.5 -w captura.pcap
```

El `.pcap` se abre luego en Wireshark.

Pro Tip:
Si no hay internet, revisa en orden: `ip a` (¿tengo IP?) → `ip r` (¿tengo gateway?) → `ping` al gateway → `ping 8.8.8.8` → `dig google.com` (¿falla el DNS?).

File System: cómo está organizado

En Linux **todo es un archivo** y todo cuelga de una sola raíz `/`. No hay letras de unidad (C:, D:): los discos se **montan** en carpetas. La estructura sigue el estándar **FHS**.

Ruta	Qué contiene
`/`	Raíz de todo el sistema
`/bin /sbin`	Comandos esenciales (hoy enlaces a /usr/bin)
`/boot`	Kernel y gestor de arranque (GRUB)
`/dev`	Dispositivos como archivos (`/dev/sda`)
`/etc`	Configuración del sistema y servicios
`/home`	Carpetas personales de los usuarios
`/root`	Home del usuario root
`/var`	Datos variables: logs, caché, bases de datos
`/tmp`	Temporales (se borra al reiniciar)
`/usr`	Programas y librerías instalados
`/opt`	Software de terceros
`/mnt /media`	Puntos de montaje manuales / USB
`/proc /sys`	Virtuales: info en vivo del kernel y procesos
`/srv`	Datos servidos (web, FTP)

Tipos de file system comunes

`ext4`	El estándar en Debian/Ubuntu. Estable, con journaling.
`xfs`	Por defecto en RHEL. Muy bueno con archivos grandes.
`btrfs`	Snapshots, compresión y subvolúmenes (Fedora, openSUSE).
`tmpfs`	Vive en RAM, desaparece al reiniciar.
`vfat / ntfs`	Compatibilidad con USB y Windows.

Journaling: registra los cambios antes de aplicarlos para recuperarse de apagones sin corromper datos. **Inodo:** ficha con los metadatos de cada archivo (dueño, permisos, bloques); el nombre solo apunta al inodo. Puedes quedarte sin inodos aunque haya espacio (`df -i`).

Discos, particiones y montaje

```
lsblk -f  # discos, particiones, FS y UUID
sudo fdisk -l
sudo mkfs.ext4 /dev/sdb1  # ⚠ formatea
sudo mount /dev/sdb1 /mnt/datos
sudo umount /mnt/datos
df -hT ; df -i            # uso por FS e inodos
sudo fsck /dev/sdb1       # reparar (desmontado)
```

Montaje automático en `/etc/fstab`

```
UUID=1234-abcd  /mnt/datos  ext4  defaults  0  2
```

Dispositivo · punto de montaje · tipo · opciones · dump · orden de fsck. Prueba con `sudo mount -a` antes de reiniciar.

Enlaces

```
ln -s /etc/nginx/sites-available/app \
      /etc/nginx/sites-enabled/
ln archivo.txt copia_dura.txt
```

`-s` Simbólico: acceso directo a una ruta (se rompe si se borra el original). Sin `-s`: enlace duro, otro nombre para el mismo inodo.