

Taller de Linux

Perdamos el miedo al terminal y a funcionar con los servidores

10 de noviembre de 2025



¿Qué vamos a ver en este taller?

1 Linux

2 Terminal

3 Comandos básicos

- pwd
- ls
- cd
- Editores de texto/código
- Vim
- Nano
- MAN

4 Comandos para comunicarnos y operar sobre servidores.

- ssh
- SCP
 - Cómo subimos un fichero al ribera
 - descargar ficheros a un servidor remoto
- ¿Cómo podemos cerrar una sesión de manera correcta?

Linux ese gran desconocido

Definición canónica

Un Sistema operativo es el principal programa que se ejecuta en un ordenador cuyo propósito es general



Figura: Las diferentes mascotas de las distros de Linux

La terminal

¿Qué es esa pestaña negra?

La *terminal* es lo que hace a Linux tan potente, puesto que es la manera más sencilla de realizar cualquier operación sobre nuestro pc



Figura: No le tengáis miedo al terminal es nuestro mejor amigo

pwd

Lista los elementos de un directorio/carpeta

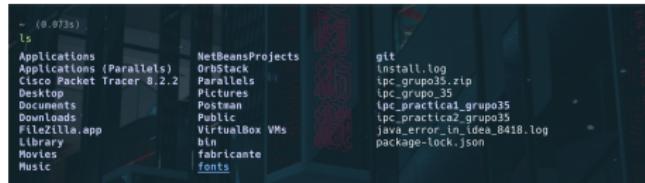
El primer comando que vamos a ver en este taller es *pwd* el cual nos permite saber donde nos encontramos ahora mismo dentro de nuestro pc



Figura: En mi caso me encuentro en el *home* del usuario *alvpi*

Lista los elementos de un directorio/carpeta

Este maravilloso comando nos permite saber qué contiene un directorio, ya sepamos su ruta absoluta o sea la carpeta donde nos encontramos.



```
~ (0.073s)
ls
Applications          NetBeansProjects
Applications (Parallels) OrbStack
Cisco Packet Tracer 8.2.2 Parallels
Desktop               Pictures
Documents              Posesos
Downloads             Public
Filezilla.app          VirtualBox VMs
Library                bin
Movies                 fabricante
Music                  fonts
```

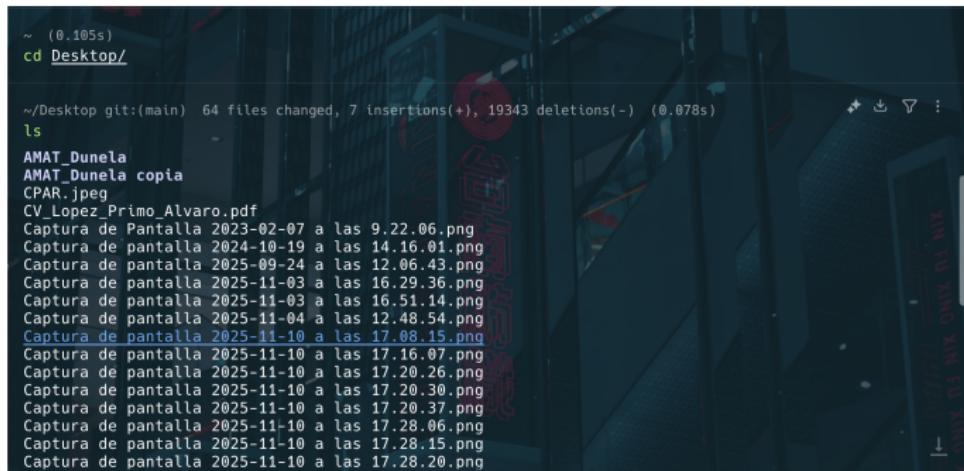
```
git
install.log
ipc_grupo35.zip
ipc_grupo39
ipc_practica4_grupo35
ipc_practica2_grupo35
java_error_in_idea_8410.log
package-lock.json
```

Figura: En la imagen se muestra todo el contenido del *home* de alvpi

cd

Desplazarnos por el directorio

Una vez que ya sabemos donde estamos y qué contienen nuestras carpetas, vamos a desplazarnos por ellas, empleando el comando que permite ejecutar un *change directory*



```
~ (0.105s)
cd Desktop/
~/Desktop git:(main) 64 files changed, 7 insertions(+), 19343 deletions(-) (0.078s)
ls
AMAT_Dunela
AMAT_Dunela copia
CPAR.jpeg
CV_Lopez_Primo_Alvaro.pdf
Captura de Pantalla 2023-02-07 a las 9.22.06.png
Captura de pantalla 2024-10-19 a las 14.16.01.png
Captura de pantalla 2025-09-24 a las 12.06.43.png
Captura de pantalla 2025-11-03 a las 16.29.36.png
Captura de pantalla 2025-11-03 a las 16.51.14.png
Captura de pantalla 2025-11-04 a las 12.48.54.png
Captura de pantalla 2025-11-10 a las 17.08.15.png
Captura de pantalla 2025-11-10 a las 17.16.07.png
Captura de pantalla 2025-11-10 a las 17.20.26.png
Captura de pantalla 2025-11-10 a las 17.20.30.png
Captura de pantalla 2025-11-10 a las 17.20.37.png
Captura de pantalla 2025-11-10 a las 17.28.06.png
Captura de pantalla 2025-11-10 a las 17.28.15.png
Captura de pantalla 2025-11-10 a las 17.28.20.png
```

Figura: En la imagen se muestra como me he desplazado a mi directorio /Desktop y como he listado el contenido del mismo

Desplazarnos por el directorio

Es el editor de código por excelencia que viene por defecto en la mayor parte de las distribuciones de Linux, el cual nos permite modificar cualquier fichero (ya sea .txt o .c)

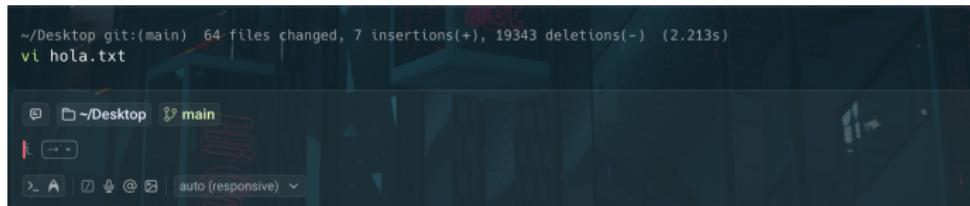
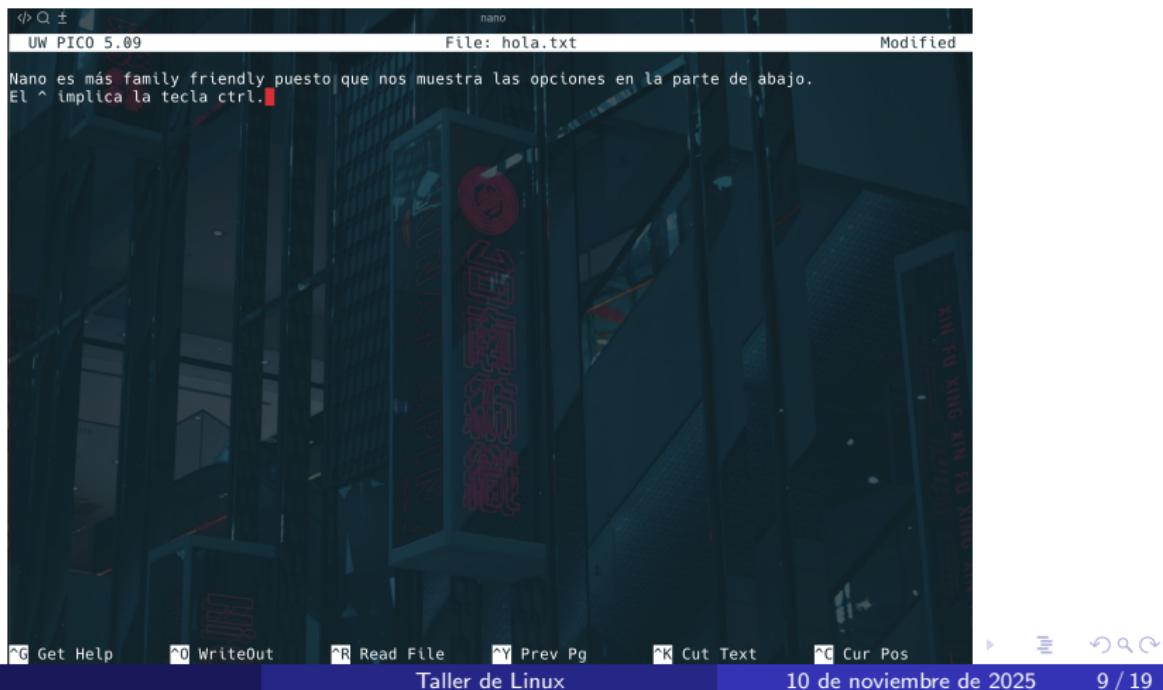


Figura: En la imagen se muestra como he empleado vi para poder crear y editar un fichero de mi */Desktop* llamado *hola.txt*

Desplazarnos por el directorio

Nano es una versión más *user-friendly* para poder realizar las mismas funciones que vi pero con una experiencia menos dura de inicio.



¿Cómo podemos memorizarnos tantos comandos y modificadores

RTFM

El manual nos permite acceder a la información de la gran variedad y herramientas de las que disponemos en nuestras máquinas con Linux

```
MKDIR(1)                                General Commands Manual                               MKDIR(1)

NAME
    mkdir - make directories

SYNOPSIS
    mkdir [-pv] [-m mode] directory_name ...

DESCRIPTION
    The mkdir utility creates the directories named as operands, in the order specified, using
    mode "rwxrwxrwx" (0777) as modified by the current umask(2).

    The options are as follows:

    -m mode          Set the file permission bits of the final created directory to the specified
                     mode. The mode argument can be in any of the formats specified to the
                     chmod(1) command. If a symbolic mode is specified, the operation characters
                     '+' and '-' are interpreted relative to an initial mode of "a=rwx".

    -p               Create intermediate directories as required. If this option is not
                     specified, the full path prefix of each operand must already exist. On the
                     other hand, with this option specified, no error will be reported if a
                     directory given as an operand already exists. Intermediate directories are
                     created with permission bits of "rwxrwxrwx" (0777) as modified by the current
                     umask, plus write and search permission for the owner.

    -v               Be verbose when creating directories, listing them as they are created.

    The user must have write permission in the parent directory.

EXIT STATUS
    The mkdir utility exits 0 on success, and >0 if an error occurs.

EXAMPLES
    Create a directory named foobar:
```

Cómo conectarnos a un servidor remoto

ssh es un comando muy potente para conexiones remotas que emplea (por defecto) el puerto 22 y es lo que entre otras muchas cosas os permite conectaros al ribera o servidores propios de la facultad de telecomunicaciones.



Figura: En la imagen se muestra como usando ssh me voy a conectar al servidor *jair* de la escuela de informática.

Secure Shell dentro del servidor

Fingerprint y conexión

Por motivos de seguridad en los servidores de la UVa se implementa un *fingerprint* el cual es un sistema que registra todas las acciones que realiza cada usuario en los mismo.

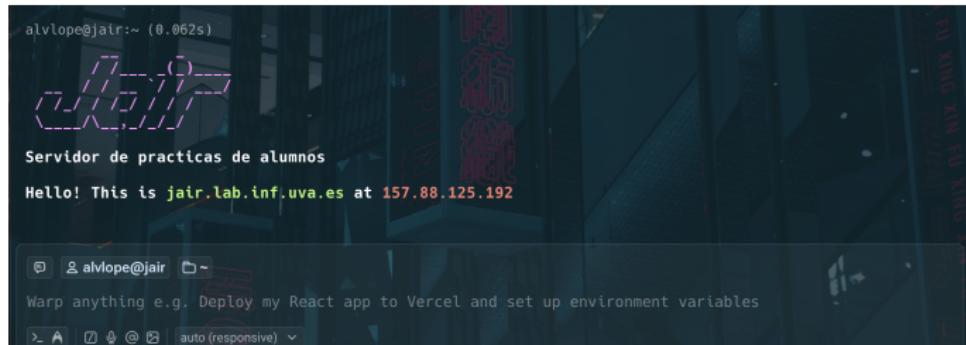


Figura: En la imagen se muestra el mensaje de inicio en jair

Ejemplo de ssh al ribera

```
jaime@JAC ~
$ ssh e[REDACTED]@ribera.tel.uva.es
e[REDACTED]@ribera.tel.uva.es's password:
Linux ribera 6.1.0-40-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.153-1 (2025-09-20) x86_64
Ribera. Acceso Restringido. Cualquier acceso puede ser monitorizado.
```

Figura: En la imagen podemos ver cómo es un intento de login exitoso en el servidor de la facultad de telecomunicaciones y el mensaje de aviso del *fingerprint*

¿Qué es?

SCP (Secure Copy) es un comando de Linux que permite transferir archivos de forma segura entre sistemas a través de SSH. Usa cifrado para copiar archivos entre equipos locales y remotos. Su sintaxis básica para subir archivos es:

```
scp archivo.txt usuario@servidor:/ruta/destino/
```

Ejercicio

Vamos a hacer un pequeño ejercicio. Cread un fichero en vuestro pc y vamos a subirlo al ribera.

Vamos a subir un archivo al ribera

```
jaime@JAC ~/Escritorio
$ scp paraMandar.txt e██████████@ribera.tel.uva.es:/home/e██████████/Escritorio/prueba.txt
e██████████@ribera.tel.uva.es's password:
paraMandar.txt                                         100%   40      2.2KB/s  00:00
```

Figura: Ejemplo de una subida de fichero exitosa

¿Cómo podemos descargar cosas del ribera?

SCP permite tanto subir como bajar archivos de un servidor. Su sintaxis básica para descargar archivos es:

```
scp usuario@servidor:/ruta/remota/archivo.txt /ruta/local/
```

Ejercicio

Segundo mini ejercicio del taller: Descargad por favor un fichero del servidor a vuestro pc.

Bajar un fichero del ribera

```
jaime@JAC ~/Escritorio
$ scp e_____@ribera.tel.uva.es:/home/e_____/Escritorio/pepe.txt /home/jaime/Escritorio/pepe.txt
e_____@ribera.tel.uva.es's password:
pepe.txt                                         100%   21      0.5KB/s  00:00

jaime@JAC ~/Escritorio
$ scp e_____@ribera.tel.uva.es:/home/e_____/Escritorio/pepe.txt /home/jaime/Escritorio/PRT2.txt
e_____@ribera.tel.uva.es's password:
Permission denied, please try again.
e_____@ribera.tel.uva.es's password:
pepe.txt                                         100%   21      0.7KB/s  00:00

jaime@JAC ~/Escritorio
$ ls
PRT2.txt  paraMandar.txt  pepe.txt

jaime@JAC ~/Escritorio
$ cat pepe.txt
pepe
texto de prueba

jaime@JAC ~/Escritorio
$ cat PRT2.txt
pepe
texto de prueba
```

Figura: Ejemplo de descargar un fichero del servidor Rivara a vuestro pc.

The Last Command

exit

Este comando nos permite cerrar una conexión ssh de manera adecuada.



```
alvlope@jair ~ (0.079s)
exit
exit
Connection to jair.lab.inf.uva.es closed.
```

Figura: Salimos de manera adecuada del jair, miso caso para el ribera.



Contacto

gui@uva.com

Github GUI

