

Comando cd (Change Directory)



El comando `cd` sirve para cambiar el directorio actual o de trabajo, desde el que voy a ejecutar comandos

1. Directorios “home”
2. `Cd` se puede usar con o sin parámetros
3. Uso de la virgullilla (~)
4. Uso del guión (-)
5. Uso de los dos puntos (..)

Nota: para descargar los ficheros de ejemplo desde el terminal de Linux, puedes ejecutar: `wget sanchezcorbalan.es/ficheros.tar.gz`

Si no tienes `wget`, puedes descargarlo desde los repositorios con: `apt update; apt install wget`

Una vez descargado el archivo *ficheros.tar.gz*, puede descomprimirlo usando `tar -xzf ficheros.tar.gz`
Esto creará un directorio llamado “ficheros” con el contenido de los ficheros de ejemplo

Comando ls (List)



El comando ls sirve para mostrar el contenido de uno o varios directorios

1. Principales opciones
 - a. -l
 - b. -h
 - c. -S
 - d. -r
 - e. -t

Permisos

Tipo de elemento



Los permisos se pueden aplicar de forma independiente según **tres conjuntos de permisos**:

- Usuario propietario
- Grupo propietario (todos los usuarios que pertenezcan a él)
- Otros. El resto de usuarios del sistema.

	R	W	X
Fichero	Leer	Escribir	Ejecutar
Directorio	Listar	Modificar	Acceder

Cambiar Permisos. chmod

Tipo de elemento



	USUARIO	GRUPO	OTROS
-	r	w	-
-	r	x	-
d	r	w	x

The diagram illustrates the structure of file permissions. It shows a 3x4 grid where the first column represents the file type (indicated by a red arrow and a red box), and the subsequent three columns represent permissions for the user (orange box), group (blue box), and others (green box). The permissions are shown as characters: 'r' for read, 'w' for write, and 'x' for execute. The first row shows the basic permissions, the second row shows the permissions with the execute bit, and the third row shows the permissions with the execute bit and the setuid/setgid/sticky bits.

El comando para cambiar los permisos de elementos del sistema de ficheros es **chmod**.

Sus parámetros son:

- Una letra para indicar el conjunto de permisos sobre el que vamos a efectuar los cambios. Si son varios se separan por comas. Si no se pone nada se realiza en todos
- Un signo de más + para añadir el permiso y uno de menos - para quitar el permiso
- La ruta del elemento sobre el que vamos a aplicar los cambios

Ejemplos

Añadir el permiso de escritura al grupo

```
chmod g+w ruta-elemento
```

Quitar el permiso de ejecución a todos

```
chmod -x ruta-elemento
```

Añadir el permiso de escritura a usuario y a otros

```
chmod u+w,o+w ruta-elemento
```

Quitar el permiso de ejecución y escritura al usuario y poner el de escritura al grupo

```
chmod u-xw,g+w ruta-elemento
```



Administración de software

Hay tres grandes formas de gestionar software (obviamos la compilación de código): los paquetes **.deb** (Debian y derivadas), **rpm** (Red Hat y derivadas) y **snap** (Ubuntu y otras)

- ¿De dónde conseguimos los ficheros) 👉 De servidores llamados REPOSITORIOS
- ¿Cómo los administramos? 👉 Debian: apt y dpkg, Red Hat: rpm y yum (o dnf)

Opciones el comando **apt**:

- **Update**: Actualiza la lista de paquetes disponibles en los repositorios.
- **Search**: Busca paquetes disponibles en los repositorios que coincidan con un término de búsqueda.
- **Show**: Muestra información detallada sobre un paquete específico.
- **List**: Lista paquetes según diferentes criterios, como los instalados, actualizables o que cumplen cierta condición de búsqueda.
- **Install**: Instala un paquete de software en el sistema.
- **Remove**: Elimina un paquete del sistema, manteniendo los archivos de configuración.
- **Purge**: Elimina un paquete del sistema junto con sus archivos de configuración.

Nota: en distribuciones antiguas puede que sea necesario utilizar los comandos apt-get y apt-cache.



Administración de software

dpkg. Administra los paquetes de nuestro sistema sin usar repositorios

- **-i** : Instala un paquete. Necesita la ruta completa del fichero .deb a instalar
`dpkg -i htop_2.0.2-1_amd64.deb`
- **-r** : Borra un paquete pero deja los ficheros de configuración
`dpkg -r htop`
- **-P** : Borra un paquete incluyendo todos los ficheros de configuración
`dpkg -P htop`
- **-s** : Muestra información y el estado de un paquete instalado
`dpkg -s htop`
- **-l** : Lista todos los paquetes que coincidan con un patrón determinado
`dpkg -l apache*`
- **-L** : Muestra todos los ficheros que ha instalado un paquete
`dpkg -L htop`
- **-S** : Muestra los paquetes que contienen ficheros que coincidan con el patrón indicado
`dpkg -S mount*`

Comandos para mostrar texto

echo: Muestra el texto que recibe.

- -e: interpreta los caracteres especiales después de \

cat : Muestra un fichero de texto por pantalla

- -n: muestra el número de línea

more: Muestra el texto poco a poco, con el espacio avanzamos una pantalla, con el intro una línea, con la q salimos y la h nos muestra ayuda

less : Parecido a more, pero más potente. Acepta los controles mencionados y también puede realizar búsquedas o modificaciones en los ficheros. Recomendable para ficheros muy grandes.

head : Muestra las N primeras líneas de uno o varios ficheros de texto. Por defecto las 10 primeras

- -n: cambia la cantidad de líneas a mostrar

```
head -n5 file.txt    //    head -5 file.txt
```

tail: Igual que head pero muestra las últimas líneas

nano: sencillo editor de texto en modo consola.



Comandos para filtrar texto

grep : Muestra sólo las líneas que cumplen con un patrón.
Se pueden usar palabras o expresiones regulares.

- **-i**: No distingue entre mayúscula y minúscula
- **-w**: El patrón tiene que ser una palabra independiente
- **-v**: Muestra las que NO coinciden con el patrón
- **-r**: Busca en los ficheros de forma recursiva
- **-n**: Indica el número de línea
- **-l**: Sólo indica el nombre del fichero donde ha encontrado alguna coincidencia
- **-c**: Muestra la cantidad de líneas que cumplen con el patrón
- **-B n**, **-A n**: Muestran una cantidad de líneas n, antes o después de encontrar el patrón
- **--color**: destaca el patrón en color dentro de la línea seleccionada

Tip: en un patrón, **^** significa el principio de la línea y **\$** el final. Ejemplo: `grep ^a /etc/passwd`



Comandos para filtrar texto

cut : Muestra sólo una parte de cada línea. Hace un “corte” vertical

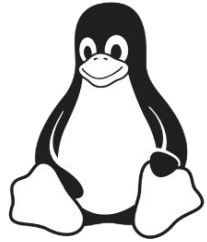
- **-c**: Selecciona sólo los caracteres que le indiquemos. Se pueden utilizar números independientes separados por comas o dos números separados por guión para indicar el inicio y el fin de un rango. Si alguno no está presente se entiende que será el inicio o fin de línea.

```
cut -c 5,10 file.txt      //      cut -c 7-25 file.txt
```

- **-d**: Indica un carácter separador entre los distintos campos de una línea. Así podremos seleccionar la información por columnas. Por defecto es el carácter tabulador
- **-f**: Elige las columnas que queremos que se muestren. La forma de seleccionar funciona igual que para la opción -c



REDIRECCIÓN



SALIDA
ESTÁNDAR



Cuando un comando produce unos resultados en modo texto, los envía a la salida estándar del sistema, que de forma predeterminada es el monitor.

`echo "HOLA MUNDO"`

HOLA MUNDO

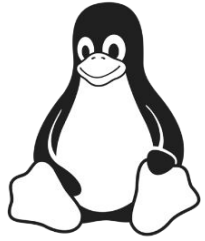


`cat /etc/passwd`

ROOT:0:0::/ROOT:
ALUMNO:5:9::/HO
TUX:8:8::/HOME



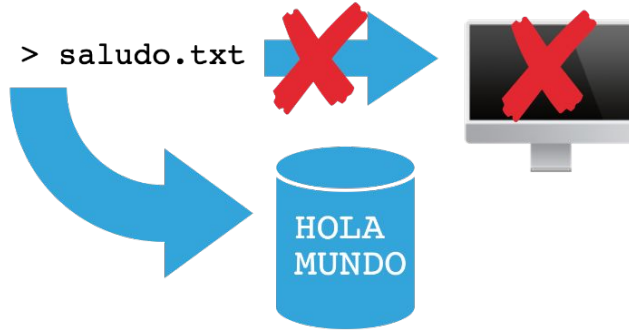
REDIRECCIÓN



Se puede enviar esa información a otro dispositivo con el carácter >

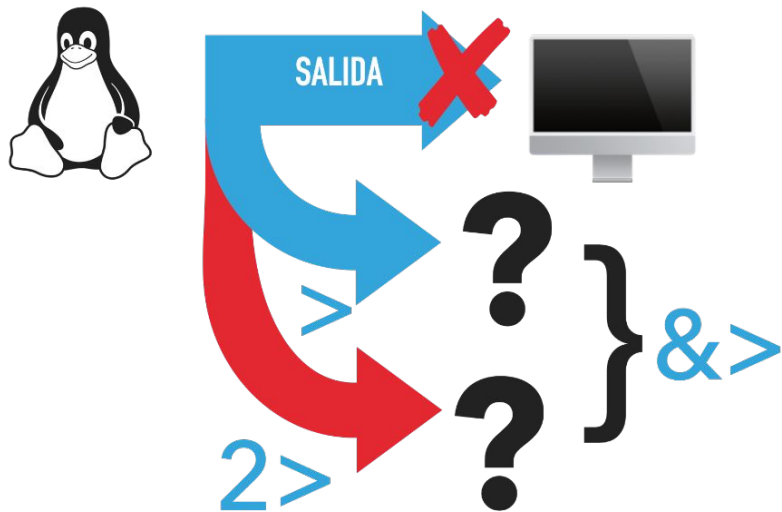
Por ejemplo, escribirla en un fichero.

```
echo "HOLA MUNDO" > saludo.txt
```

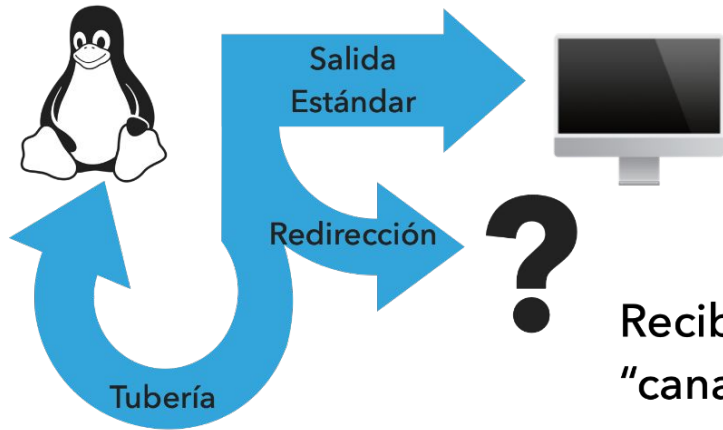


REDIRECCIÓN

- Usando un solo > si el fichero existe, lo sobrescribe
- Usando dos >> si existe añade el texto al final
- Los mensajes de error se pueden redirigir de forma independiente usando 2> o 2>>
- Para redirigir todos los mensajes se usa &> o &>>



TUBERÍAS



Además de enviar texto a un dispositivo, disponemos de otra potente opción: enviarlo a otro comando. Esto se realiza con el carácter |

Recibe el nombre de tubería, ya que establece un "canal" por el que pasará el texto de un comando a otro

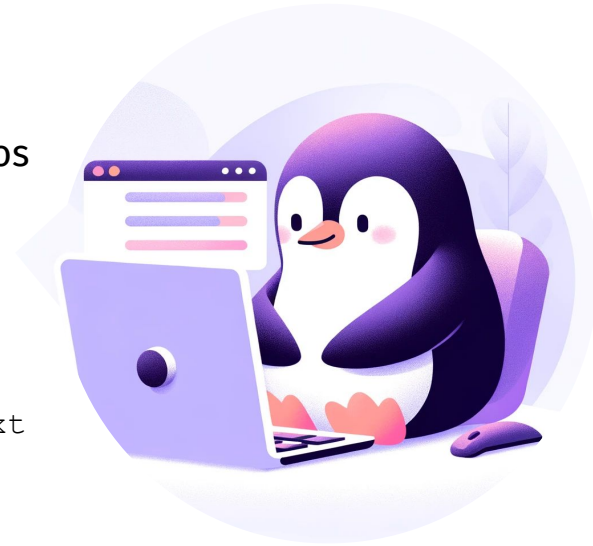
```
echo "HOLA MUNDO" | wc  
grep -w ana /etc/passwd | cut -d":" -f1 | sort
```

Sort

El comando sort **ordena** las líneas de texto en los archivos especificados o entradas estándar.

Opciones principales:

- Ordenar en Orden Inverso (**-r**)
- Ordenar Numéricamente (**-n**)
- Ordenar por Columnas Específicas (**-k**) 👉 `sort -k 2 archivo.txt`
- Especifica el Carácter de separación de columnas (**-t**)
- Eliminar Líneas Duplicadas (**-u**)



Uniq

Elimina líneas repetidas contiguas en un archivo o entrada estándar.

Opciones principales:

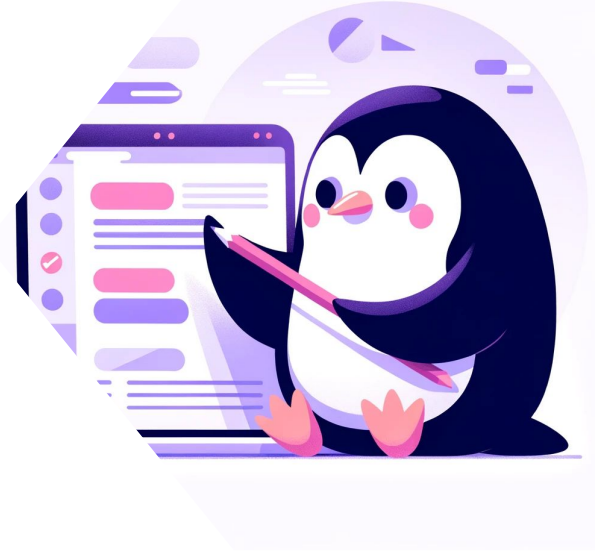
- Muestra las no duplicadas (**-u**)
- Cuenta la cantidad de repeticiones (**-c**)
- Ignora mayúsculas y minúsculas (**-i**)

WC

Se utiliza para **contar** líneas, palabras y bytes en archivos de texto o entradas estándar

Opciones principales:

- Contar Líneas (-l)
- Contar Palabras (-w)
- Contar Bytes (-c)
- Contar Caracteres (-m)



REV

Se utiliza para **invertir** el orden de los caracteres en cada línea de un archivo de texto o entrada estándar.

Es útil cuando una línea se puede dividir en columnas y quiero convertir la última en la primera.