

COMPUTER SYSTEMS  
UD4: OPERATING SYSTEMS  
FILE SYSTEMS COMMANDS

CFGS DAW  
DPT INF

## COMANDOS GESTIÓN DE ARCHIVOS Y DIRECTORIOS

\* **pwd** (print working directory)

Sintaxis: pwd

La orden pwd muestra la ruta de acceso del directorio actual.

\* **cd** (change directory)

Sintaxis: cd directory

La orden cd cambia al directorio especificado en directory

Ejemplos:

**\$cd apache** -> cambia al directorio apache que se encuentra en el directorio actual.

**\$cd /apache** -> cambia al directorio apache que se encuentra en el directorio raíz o root de la jerarquía de directorios del sistema operativo.

**\$cd /apache/web** -> cambia al directorio web situado dentro apache.

**\$cd ..** -> cambia al directorio anterior.

**\$cd (sin argumento) /\$cd ~** -> cambia al directorio personal del usuario actual.

**\$cd ~alumno** cambia al directorio personal del usuario alumno

.

\* **ls** (list)

Sintaxis: ls [options] directory

Opciones:

-A Lista todos los archivos, incluidos los ocultos (En UNIX los archivos ocultos son aquellos cuyo nombre empieza con un "."), excepto los archivos "." y ".."

-l Lista los archivos en formato largo y muestra información detallada sobre ellos.

-R Lista de forma recursiva los contenidos de los subdirectorios.

-i Muestra el número de i-node de cada fichero.

-s Muestra el tamaño en KiloBytes junto a cada archivo.

-u Clasifica por fecha y hora del último acceso.

-t Clasifica por fecha y hora de la última modificación

## COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW  
DPT INF

### \* **cp** (copy)

Sintaxis: `cp [options] file1 file2 ->` Copia file1 a file2. Si file2 existe y el usuario tiene los permisos apropiados el archivo será reemplazado.

`cp [options] files directorio ->` Copia uno o más archivos en directorio. Si no existe se mostrará un mensaje de error.

Opciones:

- f (force) Fuerza a sobrescribir los archivos existentes en el destino.
- i (interactive) Pregunta antes de sobrescribir cualquier archivo.
- p Mantiene toda la información del archivo; propietario, grupo propietario, permisos, hora y fecha. Sin esta opción, el archivo o archivos copiados tendrán la fecha y hora actual, los permisos, propietario y grupo propietario por defecto.
- R (recursive) Si en file1 se especifica un directorio, la opción `-r` o `-R` copia toda la jerarquía del directorio en el destino especificado.
- v (verbose) Muestra el nombre de cada archivo mientras se copia.

### \* **mkdir** (make directory)

Sintaxis: `mkdir [options] directory`

Crea un directorio. El usuario tiene que poseer permisos de escritura en el directorio donde se creará el directorio.

Opciones:

- p Crea los directorios intermedios si estos no existen.

### \* **mv** (move)

Sintaxis: `mv [options] source target`

Mueve o renombra archivos y directorios. Si target no existe, source es renombrado. Si target existe, será sobrescrito. Si target es un directorio, source será movido dentro de ese directorio.

Opciones:

- f Fuerza a no preguntar si el target existe, eliminando los mensajes de advertencia.
- i Fuerza a preguntar antes de mover cualquier archivo.

### \* **rm** (remove)

Sintaxis: `rm [options] files`

Elimina uno o más archivos del sistema. Para eliminar un archivo es imprescindible que el usuario tenga permiso de escritura en el directorio que contiene el archivo, pero no necesita permiso de escritura en el archivo. El comando `rm` también puede borrar directorios cuando se usan las opciones `-r` o `-R`.

Opciones:

## COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW  
DPT INF

- f Fuerza a no preguntar al borrar archivos sin permiso de escritura.
- i Fuerza a preguntar al borrar cada archivo.
- r Si file es un directorio, elimina recursivamente el contenido completo del directorio, incluidos los subdirectorios.

### \* **rmdir** (remove directory)

Sintaxis: `rmdir [options] directory`

Borra directorios vacíos.

Opciones

- p Borra los directorios intermedios si estos están vacíos como resultado de la orden.

### \* **touch**

Sintaxis: `touch [options] files`

Cambia la fecha del último acceso o/y modificación de files. Si no se especifica ninguna opción se actualizarán ambas fechas (acceso y modificación).

Opciones:

- a Actualiza únicamente la fecha del último acceso del archivo.
- m Actualiza únicamente la fecha de modificación del archivo.
- t No utiliza la fecha actual, sino el especificado a continuación mediante el formato de `[[CC]YY]MMDDhhmm[.ss]`.

Ejemplos:

Modifica la fecha del último acceso al 12 de enero de 2001 a las 18 horas, 45 minutos.

`$touch -ta 200101121845 file`

### \* **ln** (link)

Sintaxis: `ln [options] file link`

`ln [options] files directory`

Crea enlaces entre archivos. En la primera forma se crea un enlace llamado link que apunta al archivo file. En la segunda forma, se crea un enlace dentro del directorio *directory* para cada uno de los archivos especificados en files.

Opciones:

- f Fuerza a sobrescribir los enlaces si existen previamente.
- i Pregunta antes de crear cada enlace.
- s Crea un enlace simbólico. Por defecto crea enlaces hardware.

## COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW  
DPT INF

### \* **cat**

Sintaxis: cat [options] file

Muestra el contenido del archivo file.

Opciones:

- b Numera todas las líneas de salida que no están en blanco.
- n Numera todas las líneas de salida.
- s Reemplaza por una línea en blanco varias líneas en blanco adyacentes.

### \* **head /tail**

Sintaxis: head [options] file / tail [options] file

Muestra las n primeras/últimas líneas del archivo file.

Opciones:

- n Indica las n primeras/últimas líneas del archivo

### \* **uniq**

Sintaxis: uniq [options] file

Compara las líneas y busca líneas únicas. Si las líneas son iguales, muestras sólo una de ellas.

Opciones:

- n Ordena los campos numéricos por su valor numérico.
- r Realiza una ordenación inversa (de mayor a menor).

### \* **cut**

Sintaxis: cut [options] file

Recorta líneas especificando el número de caracteres:

Opciones:

- n Ordena los campos numéricos por su valor numérico.
- r Realiza una ordenación inversa (de mayor a menor).

## TUBERÍAS Y REDIRECCIONAMIENTO

Los comandos de Linux se pueden encadenar de forma que una secuencia de estos se pueda ejecutar a fin de obtener un resultado.

## TUBERÍAS

Mediante el uso de tuberías hacemos que la salida de un comando se redirija a

## COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW  
DPT INF

la entrada del siguiente:

comando1 | comando2 | comando3

### Ejemplo 1:

```
$ ls -li | cut -d" " -f1,2,41
```

Muestra información sobre el *i-nodo* de los archivos del directorio. *Cut* extrae en este caso los campos número 1, 2 y 4 del resultado:

Si introducimos sólo la orden `$ ls -li`, obtenemos:

```
total 0 1688849860306492 -rw-r--r-- 1 jose jose 0 Apr 19 12:10 archivo.txt
```

A *cut* li indicamos que queremos extraer (*recortar*) una serie de campos (-f1,2,4), los campos se indican utilizando un espacio como separador (-d" "):

```
total 0 1688849860306492 -rw-r--r-- jose
```

### Ejemplo 2:

```
$ sudo ps aux | grep init
```

Buscamos de los procesos que se están ejecutando (*ps aux*) aquellos que contienen el texto *init* (*grep init*):

```
ubuntu@ubuntu-VB:~$ sudo ps aux | grep init
[sudo] contraseña para ubuntu:
root      1  1.5  0.5 167624 11492 ?        Ss   16:50   0:02 /sbin/init splash
ubuntu   3233  0.0  0.0 11668   732 pts/0    S+   16:53   0:00 grep --color=auto init
```

## REDIRECCIONAMIENTO

Cualquier proceso tiene en Linux una entrada (**stdin**) y dos salidas (**stdout/stderr**) que se denominan **estándar**, es decir, normalmente la entrada desde el teclado y la salida al monitor, tanto para resultados como para errores, es el comportamiento habitual.

Pero también existe la posibilidad de hacerle llegar datos de entrada (parámetros) y hacer que

---

<sup>1</sup> El símbolo “|” es el que se obtiene con las teclas “Alt Gr + 1”.

## COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW  
DPT INF

la salida (resultado) sean diferentes al teclado y a la pantalla, normalmente un archivo de datos donde tendremos el resultado (error) para revisarlo.

Podemos utilizar los siguientes modificadores

Carácter	Significado
<	Cambia la entrada estándar a un archivo
>	Cambia la salida a un archivo <sup>2</sup>
>>	Com en el caso anterior, pero añade la información al final del archivo
2>	Redirecciona la salida de errores
2>>	Redirecciona los errores a un archivo, añadiendo la información

### EL COMANDO TEE

Con **tee** tenemos la ventaja de redireccionar la salida tanto a un archivo como a la pantalla, podemos tener un mejor control de la ejecución del comando.

### ARCHIVOS ESPECIALES

En Linux se llaman archivos especiales a un tipo de archivos que el sistema operativo crea y gestiona, normalmente relacionados con los dispositivos y el sistema de entrada y salida (E/S).

Ejemplos obtenidos en el directorio /dev:

```
crw--w---- 1 root tty 4, 0 d'abr. 21 15:56 tty0 ...  
brw-rw---- 1 root disk 8, 1 d'abr. 21 15:56 sda1 ...  
srw-rw-rw- 1 root root 0 d'abr. 21 15:56 log
```

Donde el primer dígito indica:

<sup>2</sup> El archivo se creará o se borrará en función de su contenido.

## COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW  
DPT INF

Caràcter	Tipo de dispositivo	Funció
c	de caracteres	Dispositivos como terminales, impresoras, etc...
b	de bloques	Para un dispositivos de bloque, como los discos duros, etc...
s	<i>sockets</i>	Tipos de archivos para comunicación en red
t	tuberías	Comunicación entre procesos

Al manipular archivos con el intérprete de comandos, a menudo es necesario realizar una determinada operación con muchos archivos. Por ejemplo, en el desarrollo de un programa en C, es necesario usar la orden touch para forzar la compilación de todos los archivos.

Para realizar este tipo de operaciones de una manera rápida y simple, el intérprete de comandos dispone de varios metacaracteres. En vez de especificar el nombre de cada archivo, los metacaracteres sustituyen parte del nombre del archivo. Los metacaracteres disponibles en GNU/LINUX se resumen en la siguiente tabla.

## COMANDOS MSDOS vs Linux

metacaràcter	descripció
*	Sustituye cualquier número de caracteres, incluido cero caracteres. Por ejemplo, x* se corresponde con los archivos o directorios x, xy, xyz, x.txt, xy.txt,...
?	Sustituye únicamente un carácter. Por ejemplo, x? se corresponde con xx, xy, xz, pero no con "x" ó "xyz".
[caracteres]	Sustituye un único carácter que este listado entre los corchetes. Por ejemplo, x[yz] se corresponde con "xy" ó "xz".
[!caracteres]	Sustituye un único carácter que <b>NO</b> este listado entre los corchetes. Por ejemplo, x[yz] se corresponde con todos los archivos o directorios cuyo segundo carácter no es ni "y" ni "z".
[a-z]	Sustituye un único carácter que este dentro del rango especificado entre los corchetes. Por ejemplo, x[0-3] se corresponde con x0, x1, x2 y x3, pero no se corresponde con "xx" ó "x4".
[!a-z]	Sustituye un único carácter que <b>NO</b> este dentro del rango especificado entre los corchetes.



## COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW  
DPT INF

DOS	Comando Linux
cd directorio	cd directorio
dir, dir/w	ls, ls -l
chdir (directorio actual)	pwd
del (borra un archivo)	rm
deltree (borra un directorio y su contenido)	rm -r
copy	cp
xcopy (copia todo el contenido de un directorio)	cp -R
rename, move	mv
type (imprime el contenido de un archivo a la pantalla)	cat
help, [comando] /?	man
cls (limpia la pantalla)	clear
find (busca por una palabra(s) en un determinado archivo)	grep
edit nombre-de-archivo	gedit nombre-de-archivo
mem (muestra la memoria disponible)	free, top
scandisk	fsck
pkzip (crea un paquete de archivos)	tar, utilizado en conjunto con gzip para compresión
ipconfig (visualiza dirección IP y configuración de red)	ifconfig
route print (muestra tablas de ruteo)	route -n

## PRÁCTICA: GESTIÓN DE ARCHIVOS

**Paso 1.** Abrir un terminal. Examinar el directorio en el cual nos ha situado el sistema por defecto. Este directorio recibe el nombre de personal y en principio es donde tenemos permisos para crear carpetas y dejar nuestros documentos.

**Paso 2.** Visualizar el contenido del directorio personal.

**Paso 3.** Acceder desde el directorio personal a los siguientes directorios

/etc /root /home /boot /bin

- de forma directa o absoluta (todo el path)

**Paso 4.** Acceder desde el directorio personal a los siguientes directorios

/etc /root /home /boot /bin

- de forma relativa

**Paso 5.** Desde cualquier lugar del sistema de directorios, acceder al directorio personal del usuario actual, utilizando el carácter ~.

**Paso 6.** Crear en el directorio personal del usuario actual los siguientes directorios: test y practica\_sin\_archivos



## COMPUTER SYSTEMS UD4: OPERATING SYSTEMS

CFGS DAW  
DPT INF

NOTA: En UNIX los nombres de archivo y directorio no pueden contener espacios en blanco.

**Paso 7.** Entrar en el directorio test y crear el directorio linux.

**Paso 8.** Situarse en el directorio personal del usuario. Con una sola orden crear el directorio test.1 y dentro de test.1 crear el directorio test.2.

**Paso 9.** Crear los siguientes archivos mediante la orden del sistema operativo ls –al > “archivo” dentro del directorio practica\_sin\_archivos.

```
test testa testA testB TestBa
testC testCa testCb testCc TestDa
testDb testDc testDd testDx TestDy
```

**Paso 10.** Cambiar el directorio actual de trabajo a test. Sin cambiar de directorio copiar todos los archivos del directorio practica\_sin\_archivos de 6 caracteres terminados en a, al directorio test.2. Indica los ficheros copiados. Especificar la opción *verbose* y *force* en el comando.

**Paso 11.** Situarse en el directorio personal del usuario actual. Copiar del directorio practica\_sin\_archivos a test.2 los archivos de 6 caracteres donde el quinto carácter no sea ni una a ni una b. Indica los ficheros copiados. Especificar la opción *interactive* de la orden.

**Paso 12.** Borra los ficheros copiados y copia sólo ahora los archivos de 5 caracteres donde el quinto carácter no sea ni una a ni una b. Indica los ficheros copiados

**Paso 13.** Listar el contenido (incluido los archivos ocultos) de los directorios practica\_sin\_archivos y test.2 sin salir del directorio personal del usuario.

**Paso 14.** Mover del directorio practica\_sin\_archivos al directorio linux todos los archivos que NO terminen en a, b, c y d usando el metacarácter de intervalo. Especificar la opción *force* y *verbose* de la orden.

**Paso 15.** Cambiar el directorio actual de trabajo a practica\_sin\_archivos. Renombrar el archivo test a test.txt. Si no existe test, crearlo mediante el editor vi.

**Paso 16.** Cambiar la fecha de modificación del archivo test.txt a 20 de diciembre de 1973 11 horas 35 minutos de la mañana.

**Paso 17.** Crear un enlace simbólico a test.txt que se llame test.txt.link.

**Paso 18.** Visualizar el fichero test.txt y test.txt.link mediante la orden cat. ¿Hay alguna diferencia al visualizar los dos archivos?

**Paso 19.** Eliminar el contenido de los directorios test, test.1 y practica\_sin\_archivos. Especificar la opción *verbose*, *interactive*, *forzado* y *recursive* del comando.