

En esta práctica veremos algunas operaciones básicas sobre archivos que nos permitirán conocer propiedades básicas de los mismos. Vamos a aprender a crear, copiar, mover, renombrar y borrar ficheros y directorios, así como empaquetar y comprimir directorios. También aprenderemos a utilizar los meta caracteres para reducir la escritura de mandatos que afectan a un conjunto de ficheros.

Durante el desarrollo de esta práctica ve elaborando un informe en el que indiques brevemente los pasos dados en su realización, y contesta a las preguntas que se te van haciendo en el guion. Entrega este informe, en formato pdf, a través del aula virtual de la asignatura.

Guión de la práctica:

1. Copiar, mover, renombrar y borrar ficheros y directorios en Linux.

a. Inicia la máquina virtual Ubuntu.

b. Abre un terminal. (Elige Aplicaciones/Accesorios/Terminal)

c. Crea un subdirectorio llamado practica5. (mkdir practica5)

```
mario@mario-VirtualBox:~$ mkdir practica5
```

d. Haz que practica5 sea el directorio de trabajo. Comprueba siempre el resultado de ejercicios con el mandato "ls" pero sin cambiar de directorio de trabajo.

```
mario@mario-VirtualBox:~$ cd practica5/  
mario@mario-VirtualBox:~/practica5$
```

f. Lee cuál es la función del mandato "touch". Se suele utilizar "touch" para crear ficheros vacíos. (man touch)

Crear un fichero de texto

g. Crea los ficheros vacio1 y vacio2. (touch vacio1 vacio2)

```
mario@mario-VirtualBox:~/practica5$ touch vacio1 vacio2  
mario@mario-VirtualBox:~/practica5$
```

i. Lee en el manual cuál es la función del mandato "cp".

```
mario@mario-VirtualBox:~/practica5$ man cp  
mario@mario-VirtualBox:~/practica5$
```

j. Copia el fichero vacio1 en otro fichero llamado vacio3. (cp vacio1 vacio3)

```
mario@mario-VirtualBox:~/practica5$ man cp  
mario@mario-VirtualBox:~/practica5$ cp vacio1 vacio3  
mario@mario-VirtualBox:~/practica5$
```

k. Crea una copia oculta del fichero vacio1 con el mismo nombre. Recuerda que los archivos ocultos en los sistemas Linux se caracterizan únicamente porque su nombre empieza por "." (cp vacio1 .vacio1)

```
mario@mario-VirtualBox:~/practica5$ cp vacio1 .vacio1
mario@mario-VirtualBox:~/practica5$
```

l. Copia el fichero vacio1 en el directorio dir2 (cp vacio1 dir2)

```
mario@mario-VirtualBox:~/practica5$ cp vacio1 dir2
mario@mario-VirtualBox:~/practica5$
```

m. Copia el fichero vacio1 en el directorio dir2 con el nombre vacio4. (cp vacio1 dir2/vacio4)

```
mkdir: cannot create directory 'dir2': File exists
mario@mario-VirtualBox:~/practica5$ cp vacio1 dir2/vacio4
mario@mario-VirtualBox:~/practica5$
```

n. Haz que dir1 sea el directorio de trabajo y crea dos ficheros de texto por medio de nano, de nombres prueba1 y prueba2. El contenido de los mismos queda a tu

elección (pero no los dejes vacíos; conviene que sean distintos). o. Copia el fichero prueba1 en el directorio dir2. Recuerda que para "subir un nivel" en el

árbol de directorios puedes utilizar la abreviatura ".." (cp prueba1 ../dir2)

```
mkdir: cannot create directory 'dir2': File exists
mario@mario-VirtualBox:~/practica5$ cp prueba1 ../dir2
mario@mario-VirtualBox:~/practica5$
```

p. Copia el fichero vacio23 del directorio dir2 al directorio dir1. Date cuenta de que el directorio de destino es el directorio de trabajo, es decir dir1. Por tanto necesitas hacer referencia a él con "." (cp ../dir2/vacio23 .)

```
mario@mario-VirtualBox:~/practica5$ cp ../dir2/vacio23 .
mario@mario-VirtualBox:~/practica5$
```

q. Copia el fichero vacio4 del directorio dir2 al directorio dir1 con el nombre vacio5.

```
mkdir: cannot create directory 'dir2': File exists
mario@mario-VirtualBox:~/practica5$ cp dir2/vacio4 ../dir1/vacio5
mario@mario-VirtualBox:~/practica5$
```

s. El mandato "cp" también nos permite copiar directorios (con sus contenidos).

Generalmente, si queremos copiar directorios queremos que se copien también todos sus ficheros y subdirectorios. Para ello debemos usar la opción "-r" (copia recursiva). Realiza una copia completa del directorio /usr/games en dir2. (cp -r /usr/games dir2)

```
mario@mario-VirtualBox:~/practica5$ cp -r /usr/games dir2
mario@mario-VirtualBox:~/practica5$
```

t. Además de copiar ficheros y directorios, también podemos mover ficheros entre directorios. Lee en el manual cuál es la función del mandato "mv".

```
mkdir: cannot create directory 'dir2': File exists
mario@mario-VirtualBox:~/practica5$ man mv
mario@mario-VirtualBox:~/practica5$
```

u. Mueve el fichero prueba2 del directorio dir1 a dir2. (mv dir1/prueba2 dir2). Mueve el fichero prueba2 del directorio dir2 a practica5.

```
mkdir: cannot create directory 'dir2': File exists
mario@mario-VirtualBox:~/practica5$ mv dir 1/prueba2 dir2
```

w. Lee en el manual cuál es la función del mandato "rm".

```
mkdir: cannot create directory 'dir2': File exists
mario@mario-VirtualBox:~/practica5$ man rm
```

x. Borra el fichero vacio1 (rm vacio1). Borra algún fichero de los directorios dir\_renombrado y dir2.

```
mkdir: cannot create directory 'dir2': File exists
mario@mario-VirtualBox:~/practica5$ rm vacio1
mario@mario-VirtualBox:~/practica5$
```

## 2. Sustitución de caracteres con metacaracteres o comodines \*, ? y rangos [...]

Los metacaracteres o comodines representan a un conjunto de caracteres y sirven para abreviar mandatos.

El “\*” representa a cualquier número de caracteres (incluido 0),

el “?” representa a cualquier carácter (pero sólo un único carácter) y los

“[...]” también representan un único posible carácter entre los que aparecen dentro de los corchetes.

Veremos cómo se usan con el mandato “echo” pero se pueden utilizar siempre que queramos referirnos a varios ficheros en mandatos como “cp”, “mv”, “rm”, etc.

a. Crea los siguientes archivos desde el directorio practica5.

hola. hola a b c fich1 fich\_a FICH\_A h1dir/pdato h1dir/fich.dos h2dir/pfich ab1.fich 1datos 2datos 3datos 2DATOS.apunto, fich\_b cap1.dat cap2.txt cap31.txt capb.ff (Utiliza el mandato touch para crearlos todos en una única instrucción)

```
mario@mario-VirtualBox:~/practica5$ touch hola .hola a b c fich1 fich_a h1dir/pdato h1dir/fich2 h2dir/pfich abd1 fich 1datos 2datos 3datos 2DATOS.apunto fich_b cap1.dat cap2.txt cap31.txt capb.ff
```

b. Muestra los nombres de todos los ficheros y directorios (no ocultos) del directorio de trabajo. El “\*” no sustituye al punto inicial (echo \*)

```
mario@mario-VirtualBox:~/practica5$ echo *
1datos 2datos 2DATOS.apunto 3datos a abd1 b c cap1.dat cap2.f
f dir2 fich fich1 fich_a fich_b hola vacio2 vacio3
```

c. Muestra los nombres de todos los ficheros y directorios ocultos, es decir, que empiecen por ‘.’. (echo .\*)

```
mario@mario-VirtualBox:~/practica5$ echo .*
. . . .hola .vacio1
```

d. Muestra los nombres de ficheros y directorios formados por un único carácter (distinto de ‘.’). (echo ?)

```
mario@mario-VirtualBox:~/practica5$ echo ?
a b c
```

e. Muestra los nombres de ficheros y directorios que comiencen por un “1” o “2”. (echo [12]\*)

```
mario@mario-VirtualBox:~/practica5$ echo [12]*  
1datos 2datos 2DATOS.apunto
```

f. Muestra los nombres de ficheros y directorios que comiencen por un dígito. (echo [0-9]\*)

```
mario@mario-VirtualBox:~/practica5$ echo [0-9]*  
1datos 2datos 2DATOS.apunto 3datos
```

g. Muestra los nombres de ficheros y directorios que contengan la letra ‘a’. (Incluso los que comienzan por ‘.’). (echo \*a\*.\*a\*)

```
mario@mario-VirtualBox:~/practica5$ echo *a*.*a*  
cap1.dat
```

h. Muestra los nombres de ficheros y directorios que comiencen por cap, después una letra o dígito, a continuación un ‘.’ y, por último, cualquier cosa. (echo cap[a-zAÑñ0-9].\*)

```
mario@mario-VirtualBox:~$ echo cap [a-zAÑñ0-9].*
```

i. Muestra los nombres de ficheros y directorios que comiencen por cualquier carácter y contengan un dígito en cualquier posición.

```
cap [a-zAÑñ0-9].*  
mario@mario-VirtualBox:~$ echo *[0-9]*  
practica5
```

### 3. Empaquetado y desempaquetado

a. Por último, vamos a ver algunos mandatos de empaquetado y desempaquetado propios de Linux. Un programa muy utilizado para empaquetar archivos y carpetas es “tar”. Puedes encontrar información sobre el mismo en <http://es.wikipedia.org/wiki/Tar>.

En realidad “tar”, por defecto, no comprime información, simplemente la empaqueta o junta en un solo fichero. Ejecuta en la shell el mandato “man tar”.

b. Vamos a empaquetar ahora los contenidos de la carpeta practica5. Dirígete antes a tu directorio \$HOME. (tar -cvf practica5.tar practica5)

```
practica5/$  
mario@mario-VirtualBox:~$ tar -cvf practica5.tar practica5
```

c. Apunta lo que hace cada una de las opciones “c”, “v”, “f”. Ejecuta ahora “ls -l” y comprueba el nuevo archivo que ha aparecido. Los ficheros producidos por “tar” tienen algunas características importantes, como que mantienen los permisos de los archivos originales, preservan los enlaces simbólicos o fuertes, mantienen el árbol de directorios... Por eso constituye una herramienta ideal para trasladar información entre máquinas Linux.

d. Para poder descomprimir un archivo tar hay que realizar una acción parecida. Borrarnos en primer lugar la carpeta practica5. Para ello debes ejecutar “rm -r practica5”. Comprueba que has obtenido el resultado deseado (la carpeta debe haber desaparecido). Ahora vamos a desempaquetar el archivo practica5.tar: (tar -xvf practica5.tar)

```
mario@mario-VirtualBox:~$ tar -xvf practica5.tar
```

e. Apunta el significado de la opción “x”. Comprueba el resultado por medio de “ls -l”. Además de utilidades para empaquetar, pero que no reducen ni disminuyen el tamaño de los archivos originales, el intérprete de mandatos también nos presta ciertas aplicaciones que permiten comprimir carpetas o ficheros. Dos de las más conocidas son “bzip2” y “gzip”. “bzip2” consigue mayores niveles de compresión, pero también requiere un mayor uso de CPU. El uso de ambas es similar. El uso combinado de tar y bzip2 o gzip también es posible. En general es bastante común que encuentres ficheros para programas o aplicaciones que tengan extensión

“tar.gz” o “tar.bz2”, y que deberías ser capaz de descomprimir con los siguientes mandatos anteriores. Veamos cómo podemos hacerlo (puedes encontrar las

opciones combinadas correspondientes en <http://es.wikipedia.org/wiki/Tar>)

f. Apunta el tamaño del fichero practica5.tar (ejecuta “ls -l” y mira la columna correspondiente).

```
mario@mario-VirtualBox:~$ ls -l practica5.tar
-rw-rw-r-- 1 mario mario 20480 Apr  9 16:57 practica5.tar
mario@mario-VirtualBox:~$
```

g. Comprime el fichero practica5.tar. (gzip practica5.tar)

```
-rw-rw-r-- 1 mario mario 20480 Apr  9 16:57 p
mario@mario-VirtualBox:~$ gzip practica5.tar
mario@mario-VirtualBox:~$
```

h. Comprueba el tamaño del archivo obtenido (“ls -l”).

```
mario@mario-VirtualBox:~$ ls -l
total 52
drwxr-xr-x 2 mario mario 4096 Mar 31 16:04 Desktop
drwxr-xr-x 2 mario mario 4096 Mar 31 16:04 Documents
drwxr-xr-x 2 mario mario 4096 Mar 31 16:04 Downloads
-rw-r--r-- 1 mario mario 8445 Mar 31 12:38 examples.desktop
drwxr-xr-x 2 mario mario 4096 Mar 31 16:04 Music
drwxr-xr-x 2 mario mario 4096 Mar 31 16:04 Pictures
drwxrwxr-x 2 mario mario 4096 Apr  9 16:46 practica5
-rw-rw-r-- 1 mario mario 478 Apr  9 16:57 practica5.tar.gz
drwxr-xr-x 2 mario mario 4096 Mar 31 16:04 Public
drwxr-xr-x 2 mario mario 4096 Mar 31 16:04 Templates
drwxr-xr-x 2 mario mario 4096 Mar 31 16:04 Videos
```

i. Descomprímelo (gunzip practica5.tar.gz)

```
mario@mario-VirtualBox:~$ gunzip practica5.tar.gz
```

j. Vuelve a comprimirlo con la opción “-9”. La opción “-9” le indica al programa que trate de conseguir el máximo nivel de compresión (gzip -9 practica5.tar). Comprueba su tamaño. Descomprímelo. El funcionamiento de bzip2 es similar.

```
mario@mario-VirtualBox:~$ gunzip practica5.tar.gz  
mario@mario-VirtualBox:~$ gzip -9 practica5.tar  
mario@mario-VirtualBox:~$
```

k. Recupera los mandatos que has usado hasta ahora en la sesión por medio de “history” y guárdalos en un archivo como has hecho en prácticas anteriores.