

## **PRACTICA 2**

**Ejercicio 2.1.** Se debe utilizar solamente una vez la orden chmod en cada apartado. Los cambios se harán en un archivo concreto del directorio de trabajo (salvo que se indique otra cosa). Cambiaremos uno o varios permisos en cada apartado (independientemente de que el archivo ya tenga o no dichos permisos) y comprobaremos que funciona correctamente:

Abrimos la terminal con “ctrl + alt + t” y nos dirigimos al directorio en el que se encuentre el archivo con el cual queremos trabajar, en mi caso “cd Escritorio”.

- **Dar permiso de ejecución al “resto de usuarios”.**

Utilizamos la orden “chmod”, “o” para referirnos al resto de usuarios, “+” para otorgar permisos y “x” para referirnos al permiso de ejecución. A continuación, el archivo en el que queremos dar esos permisos.

```
~/Escritorio$ chmod o+x text
```

- **Dar permiso de escritura y ejecución al “grupo”.**

Utilizamos la orden “chmod”, “g” para referirnos al grupo, “+” para otorgar permisos y “w” para referirnos al permiso de escritura y “x” para el de ejecución. A continuación, el archivo en el que queremos dar esos permisos.

```
~/Escritorio$ chmod g+wx text
```

- **Quitar el permiso de lectura al “grupo” y al “resto de usuarios”.**

Utilizamos la orden “chmod”, “g” para referirnos al grupo, “-” para quitar permisos y “r” para referirnos al permiso de lectura. A continuación, el archivo en el que queremos quitar esos permisos.

```
~/Escritorio$ chmod g-rwx text
```

- **Dar permiso de ejecución al “propietario” y permiso de escritura al “resto de usuarios”.**

Utilizamos la orden “chmod”, “u” para referirnos al propietario, “+” para otorgar permisos y “x” para referirnos al permiso de ejecución. A continuación, usamos “,” para mandar otra orden chmod distinta. En este caso, usamos “o” para referirnos al resto de usuarios, “+” para

otorgar permisos y “w” para referirnos al permiso de escritura. Posteriormente, el archivo en el que queremos dar esos permisos.

```
~/Escritorio$ chmod u+w,o+w text
```

- **Dar permiso de ejecución al “grupo” de todos los archivos cuyo nombre comience con la letra “e”. Nota: Si no hay más de dos archivos que cumplan esa condición, se deberán crear archivos que empiecen con “e” y/o modificar el nombre de archivos ya existentes para que cumplan esa condición.**

Utilizamos la orden “chmod”, “g” para referirnos al grupo, “+” para otorgar permisos y “w” para referirnos al permiso de escritura. A continuación, “e\*” para referirnos a todos los archivos que empiecen por e.

```
~/Escritorio$ chmod g+w e*
```

**Ejercicio 2.2.** Utilizando solamente las órdenes de la práctica anterior y los metacaracteres de redirección de salida:

- **Cree un archivo llamado ej31 , que contendrá el nombre de los archivos del directorio padre del directorio de trabajo.**

Usamos “ls > ej31” para crear el archivo, pues en mi caso no existe ningún archivo con tal nombre en el directorio “Práctica2”. Con este paso, a su vez, conseguimos redirigir la salida estándar a “ej31”. Posteriormente, usamos la orden “ls .. > ej31” para redirigir la salida de “ls ..” (lista de los archivos encontrados en el directorio padre” a nuestro archivo. Comprobamos que hemos realizado bien la operación al usar la orden “cat ej31” y ver que, efectivamente, nos muestra los archivos del directorio padre.

```
~/FS/Práctica2$ ls > ej31
```

```
~/FS/Práctica2$ ls .. > ej31
```

```
~/FS/Práctica2$ cat ej31
```

Módulo.I.Órdenes.Unix.y.Shell.Bash.pdf

Práctica1

Práctica2

- **Cree un archivo llamado ej32 , que contendrá las dos últimas líneas del archivo creado en el ejercicio anterior.**

Creamos el archivo mediante la orden:

```
~/FS/Práctica2$ touch ej32
```

A continuación, mediante la orden:

```
~/FS/Práctica2$ tail -2 ej31 > ej32
```

Conseguimos lo que buscamos, que el archivo ej32 contenga las dos últimas líneas de ej31.

Comprobamos que la operación ha sido exitosa:

```
~/FS/Práctica2$ cat ej32
```

Práctica1

Práctica2

- **Añada al final del archivo ej32 , el contenido del archivo ej31 .**

Será tan simple como mostrar la salida de “cat ej31” en “ej32” sin que borre lo que ya estaba contenido en “ej32”, es decir, haciendo uso de “>>”.

```
~/FS/Práctica2$ cat ej31 >> ej32
```

```
~/FS/Práctica2$ cat ej32
```

Práctica1

Práctica2

Módulo.I.Órdenes.Unix.y.Shell.Bash.pdf

Práctica1

Práctica2

**Ejercicio 2.3.** Utilizando el metacarácter de creación de cauces y sin utilizar la orden cd:

- **Muestre por pantalla el listado (en formato largo) de los últimos 6 archivos del directorio /etc.**

Usamos “ls -l /etc” para mostrar el listado en formato largo de los archivos del directorio “/etc”, sin embargo, mediante un cauce, conseguimos que dichos archivos sólo sean los 6 últimos.

```
~/FS/Práctica2$ ls -l /etc | tail -6
```

- **La orden wc muestra por pantalla el número de líneas, palabras y bytes de un archivo (consulta la orden man para conocer más sobre ella). Utilizando dicha orden, muestre por pantalla el número de caracteres (sólo ese número) de los archivos del directorio de trabajo que comiencen por los caracteres “e” o “f”.**

Dentro de nuestro directorio de trabajo, hacemos uso de “cat” y “[ef]\*” para mostrar el contenido de todos los archivos que comiencen por “e” o “f”. No obstante, gracias al uso de un cauce, conseguimos que solo se muestre el nº de caracteres que dichos archivos contienen en total mediante la orden “wc -c” (-c para que solo se muestren los caracteres).

```
~/Escritorio$ cat [ef]* | wc -c  
5371
```

Si no se nos pidiese expresamente que usasemos un cauce, podríamos conseguir el mismo resultado directamente de la siguiente forma:

```
~/Escritorio$ wc -c [ef]*
```

#### **Ejercicio 2.4. Resuelva cada uno de los siguientes apartados.**

**a) Cree un archivo llamado ejercicio1, que contenga las 17 últimas líneas del texto que proporciona la orden man para la orden chmod (se debe hacer en una única línea de órdenes y sin utilizar el metacarácter “;” ).**

Utilizamos un cauce y el metacarácter “>” para poder crear el archivo ejercicio1, el cual era inexistente, y se muestre en el las 17 últimas líneas de man chmod:

```
~/Escritorio$ man chmod | tail -17 > ejercicio1  
~/Escritorio$ cat ejercicio1  
REPORTING BUGS  
  GNU coreutils online help: <http://www.gnu.org/software/coreutils/>  
  Report chmod translation bugs to <http://translationproject.org/team/>
```

#### **COPYRIGHT**

```
Copyright © 2017 Free Software Foundation, Inc.  License  GPLv3+: GNU  
GPL version 3 or later <http://gnu.org/licenses/gpl.html>.  
This is free software: you are free to change and redistribute it.  
There is NO WARRANTY, to the extent permitted by law.
```

#### **SEE ALSO**

```
chmod(2)
```

```
Full documentation at: <http://www.gnu.org/software/coreutils/chmod>  
or available locally via: info '(coreutils) chmod invocation'
```

GNU coreutils 8.28

January 2018

CHMOD(1)

**b) Al final del archivo ejercicio1, añade la ruta completa del directorio de trabajo actual.**

Hacemos uso de la orden “pwd” y del metacarácter “>” para que de esta forma no se escriba al final del archivo y no se elimine lo que ya contenía anteriormente.

```
~/Escritorio$ pwd >> ejercicio1
```

Comprobamos que la operación se haya realizado con éxito:

```
~/Escritorio$ cat ejercicio1
```

REPORTING BUGS

GNU coreutils online help: <<http://www.gnu.org/software/coreutils/>>

Report chmod translation bugs to <<http://translationproject.org/team/>>

COPYRIGHT

Copyright © 2017 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later <<http://gnu.org/licenses/gpl.html>>.

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.

SEE ALSO

chmod(2)

Full documentation at: <<http://www.gnu.org/software/coreutils/chmod>>  
or available locally via: info '(coreutils) chmod invocation'

GNU coreutils 8.28

January 2018

CHMOD(1)

/home/bece/Escritorio

**c) Usando la combinación de órdenes mediante paréntesis, cree un archivo llamado ejercicio3 que contendrá el listado de usuarios conectados al sistema (orden who) y la lista de archivos del directorio actual.**

Usamos “(who;ls)” para realizar ambas ordenes y lo mostramos en un archivo inexistente, “ejercicio3”, a partir del metacarácter “>”, el cual también crea el archivo si este no existe.

```
~/Escritorio$ (who;ls) > ejercicio3
```

Comprobamos que la operación se haya realizado con éxito:

```
~/Escritorio$ cat ejercicio3
```

bece :0 2018-09-27 11:33 (:0)

ej1FS

ej6FS

ej7FS

ejercicio1

ejercicio3  
fs  
Práctica 1 - FS.odt  
PRÁCTICA 1 - FS.pdf  
text

**d) Añada, al final del archivo ejercicio3, el número de líneas, palabras y caracteres del archivo ejercicio1. Asegúrese de que, por ejemplo, si no existiera ejercicio1, los mensajes de error también se añadieran al final de ejercicio3.**

Usamos la orden “wc ejercicio1” para saber el número de líneas, palabras y caracteres de dicho archivo. A continuación, usamos “&>>” para que se muestre la salida estándar y, en caso de error, la salida error, sin que se borre los contenidos previos en el archivo “ejercicio3”.

```
~/Escritorio$ wc ejercicio1 &>> ejercicio3
bece@bece-G5-5587:~/Escritorio$ cat ejercicio3
bece :0      2018-09-27 11:33 (:0)
ej1FS
ej6FS
ej7FS
ejercicio1
ejercicio3
fs
Práctica 1 - FS.odt
PRÁCTICA 1 - FS.pdf
text
18 74 739 ejercicio1
```

**e) Con una sola orden chmod, cambie los permisos de los archivos ejercicio1 y ejercicio3, de forma que se quite el permiso de lectura al “grupo” y se dé permiso de ejecución a las tres categorías de usuarios.**

Usamos “g” para referirnos al grupo, “-” para quitarle permiso y “r” para que dicho permiso sea el de lectura. A continuación, separamos con “,” para poder meter otra orden chmod distinta. En dicha nueva orden usamos “a” para referirnos a todos los usuarios, que incluye las tres categorías de usuario, “+” para otorgar permisos y “x” para que dicho permiso sea el de ejecución. Posteriormente, indicamos sobre que archivos queremos quitar y dar dichos permisos, es decir, ejercicio1 y ejercicio3 separados por un espacio.

```
~/Escritorio$ chmod g-r,a+x ejercicio1 ejercicio3
```

*Realizado por Alejandro Becerra Burgos - 1º DGIIM Grupo 1 de FS*