

Supuestos prácticos. Bloque Comandos SSOO Linux.

Nivel Básico

1. Listar todos los archivos del directorio bin.

ls /bin

2. Listar todos los archivos del directorio tmp.

ls /tmp

3. Listar todos los archivos del directorio etc que empiecen por t en orden inverso.

ls -r /etc/t*

4. Listar todos los archivos del directorio dev que empiecen por tty y tengan 5 caracteres.

ls /dev/tty??

5. Listar todos los archivos del directorio dev que empiecen por tty y acaben en 1,2,3 ó 4.

ls /dev/tty*[1-4]

6. Listar todos los archivos del directorio dev que empiecen por t y acaben en C1.

ls /dev/t*c1 # no hay ficheros con ese patrón, si probamos que acaben en 0 sería así: ls /dev/t*0

7. Listar todos los archivos, incluidos los ocultos, del directorio raíz.

ls -a /

8. Listar todos los archivos del directorio etc que no empiecen por t.

ls -d /etc/[!t]*

9. Listar todos los archivos del directorio usr y sus subdirectorios.

ls -R /usr

10. Cambiarse al directorio tmp y crear directorio PRUEBA.

cd /tmp | mkdir -pv PRUEBA

#si ponemos | → el segundo comando se ejecuta con el resultado de la ejecución del primero

#si ponemos ; → los comandos se ejecutan independientemente en paralelo

si ponemos || → Si falla el primer comando se ejecuta el segundo comando

si ponemos && → El segundo comando se ejecutará si no falla el primer comando, en caso de fallo, el segundo comando no se ejecuta.

la opción -p crea un directorio de forma recursiva y no lo crea si ya existe; -v nos muestra información del proceso de creación y si quisiéramos asignar permisos usamos -m

Por ejemplo: **mkdir -pv aa/bb/cc/dd/ee ; mkdir -m 700 prueba**

11. Verificar que el directorio actual ha cambiado.

pwd

12. Mostrar el día y la hora actual.

date

13. Con un solo comando posicionarse en el directorio \$HOME.

cd /home

14. Verificar que se está en él.

pwd

15. Listar todos los ficheros del directorio HOME mostrando su número de inodo.

ls -li

16. Borrar todos los archivos y directorios visibles de vuestro directorio PRUEBA.

rm -rf PRUEBA/*

17. Crear los directorios dir1, dir2 y dir3 en el directorio PRUEBA. Dentro de dir1 crear el directorio dir11. Dentro del directorio dir3 crear el directorio dir31. Dentro del directorio dir31, crear los directorios dir311 y dir312.

mkdir -pv PRUEBA/{dir1/dir11,dir2,dir3/dir31/{dir311,dir312}}

18. Copiar el archivo /etc/mstab a un archivo llamado mensaje de vuestro directorio PRUEBA.

touch PRUEBA/mensaje && cat /etc/mstab>PRUEBA/mensaje

19. Copiar mensaje en dir1, dir2 y dir3.

cd PRUEBA

cp mensaje dir1/mensaje && cp mensaje dir2/mensaje && cp mensaje dir3/mensaje

20. Comprobar el ejercicio anterior mediante un solo comando.

ls -R

#si ya estamos en PRUEBA, sino ls -R PRUEBA

21. Copiar los archivos del directorio rcS.d que se encuentra en /etc al directorio dir31.

cp -r /etc/rcS.d/ dir3/dir31

#suponemos que estamos en PRUEBA

22. Copiar en el directorio dir311 los archivos de /bin que tengan una a como segunda letra y su nombre tenga cuatro letras.

cp -r /bin/?a??? dir3/dir31/dir311

#suponemos que estamos en PRUEBA

23. Copiar el directorio de otro usuario y sus subdirectorios debajo de dir11 (incluido el propio directorio).

sudo cp -r ../../usuario2 dir1/dir11

#suponemos que estamos en PRUEBA

#debemos tener permisos de administrador para copiar ficheros de los que no somos propietarios. Por eso usamos sudo delante del comando

24. Mover el directorio dir31 y sus subdirectorios debajo de dir2.

mv dir3/dir31/* dir2

#suponemos que estamos en PRUEBA

25. Mostrar por pantalla los archivos ordinarios del directorio HOME y sus subdirectorios.

ls -R \$HOME

26. Ocultar el archivo mensaje del directorio dir3.

mv dir3/mensaje dir3/.mensaje

27. Borrar los archivos y directorios de dir1, incluido el propio directorio.

```
sudo rm -rf dir1 # usamos sudo porque tenemos ficheros de otro usuario.
```

28. Copiar al directorio dir312 los ficheros del directorio /dev que empiecen por t, acaben en una letra que vaya de la a a la b y tengan cinco letras en su nombre.

```
sudo cp -rv /dev/t???[a-b] PRUEBA/dir2/dir312
```

Esto no nos va a funcionar porque no hay ningún elemento que cumpla ese patrón, pero si los siguiente:

```
sudo cp -rv /dev/t???[0-9] PRUEBA/dir2/dir312
```

opciones: -r para copiar recursivamente y -v para mostrar lo que se va haciendo.

29. Borrar los archivos de dir312 que no acaben en b y tengan una q como cuarta letra.

```
rm -r PRUEBA/dir2/dir312/???q[^b]
```

#No funciona porque no existe ningún fichero que cumpla ese patrón, pero si lo siguiente: `rm -r PRUEBA/dir2//dir312/???S[^3]`

borraría todos los ficheros que no acaben en 3 y tengan como 4ª letra una S.

30. Mover el directorio dir312 debajo de dir3.

```
mv PRUEBA/dir2/dir312 PRUEBA/dir3/dir31
```

31. Crear un enlace simbólico al directorio dir1 dentro del directorio dir3 llamado enlacedir1.

```
ln -s /home/lorena/PRUEBA/dir1 PRUEBA/dir3/enlacedir1
```

32. Posicionarse en dir3 y, empleando el enlace creado en el ejercicio anterior, crear el directorio nuevo1 dentro de dir1.

```
cd dir3
```

```
mkdir enlacedir1/nuevo1/
```

33. Utilizando el enlace creado copiar los archivos que empiecen por u del directorio /bin en directorio nuevo1.

```
cp -r /bin/u* enlacedir1/nuevo1/
```

34. Crear dos enlaces duros del fichero mensaje, llamarlo enlace, en los directorios dir1 y dir2.

```
ln mensaje dir1/enlace; ln mensaje dir2/enlace
```

35. Borrar el archivo mensaje y copiar enlace en dir3.

```
rm mensaje ; cp dir1/enlace dir3/
```

36. Crear un enlace simbólico (llamado enlafich1) al fichero enlace de dir2 en dir1.

```
ln -s /home/ubuntu/PRUEBA/dir2 /home/ubuntu/PRUEBA/dir1/enlafich1
```

37. Posicionarse en dir1 y, mediante el enlace creado, copiar el archivo mensaje dentro de dir312.

```
cd dir1 && cp -r enlafich1 ../dir3/dir312/mensaje
```

38. Seguir en dir1 y, mediante el enlace creado, sacar por pantalla las líneas que tiene el archivo fich1.

```
cat enlafich1
```

39. Borrar el fichero fich1 de dir2.

```
rm dir2/mensaje
```

40. Borrar todos los archivos y directorios creados durante los ejercicios.

```
rm -r *
```

- 41. Crear el directorio dir2 y dir3 en el directorio PRUEBA ¿Cuáles son los actuales permisos del directorio dir2?**

```
mkdir -p dir1 dir2 |ls -l
```

- 42. Utilizando la notación simbólica, eliminar todos los permisos de escritura(propietario, grupo, otros) del directorio dir2.**

```
chmod 555 dir2
```

- 43. Utilizando la notación octal, eliminar el permiso de lectura del directorio dir2, al resto de los usuarios.**

```
chmod 551 dir2
```

- 44. ¿Cuáles son ahora los permisos asociados a dir2?**

```
ls -l
```

- 45. Crear bajo dir2, un directorio llamado dir21.**

```
mkdir dir2/dir21
```

```
#mkdir: no se puede crear el directorio «dir2/dir21»: Permiso denegado
```

- 46. Concederse a sí mismo permiso de escritura en el directorio dir2 e intentar de nuevo el paso anterior.**

```
chmod 751 dir2 mkdir dir2/dir21
```

- 47. ¿Cuáles son los valores por omisión asignados a los archivos?**

```
ls -l dir2
```

- 48. Cambiar el directorio actual al directorio dir2. Imprimir su trayectoria completa para verificar el cambio.**

```
cd dir2 && ls -lR
```

- 49. ¿Cuáles son los permisos asignados en su momento a este directorio?**

```
ls -l
```

- 50. Reiniciar el ordenador.**

```
reboot
```

- 51. Crear cuatro nuevos directorios llamados dira, dirb, dirc, y dird bajo el directorio actual.**

```
mkdir dira dirb dirc dird
```

- 52. Comprobar los permisos de acceso de los directorios recién creados para comprobar el funcionamiento del comando umask.**

```
ls -l
```

- 53. Crear el fichero uno . Quitar todos los permisos de lectura. Comprobarlo. Intentar borrar dicho fichero.**

```
touch uno && chmod a-r uno && ls -l && rm uno && ls -l
```

- 54. Quitarle todos los permisos de paso al directorio dir2 y otorgarle todos los demás.**

```
chmod = dir2 && chmod o=rwx dir2
```

- 55. Crear en el directorio propio:**

- a. *El directorio carpeta1 con los tres permisos para el propietario, dentro de él fich1 con lectura y escritura para todos y fich2 con lectura y escritura*
- b. *para el propietario y solo lectura para el resto.*
- c. *El directorio carpeta2 con todos los permisos para el propietario y lectura y ejecución para los del mismo grupo. Dentro file1 con lectura y escritura para el propietario y los del grupo y file2 con los mismos para el propietario y solo lectura para el grupo.*

```
mkdir -m 700 carpeta1 && touch carpeta1/fich1 && touch carpeta1/fich2 && chmod 666 carpeta1/fich1 && chmod 644 carpeta1/fich2
```

```
mkdir -m 750 carpeta2 && touch carpeta2/fich1 && touch carpeta2/fich2 && chmod 660 carpeta2/fich1 && chmod 640 carpeta2/fich2
```

- 56. Desde otro usuario probar todas las operaciones que se pueden hacer en los ficheros y directorios creados.**

```
ls -lR
```

- 57. Visualizar la trayectoria completa del directorio actual. Crear dos directorios llamados correo y fuentes debajo del directorio actual.**

```
mkdir correo fuentes
```

- 58. Posicionarse en el directorio fuentes y crear los directorios dir1, dir2, dir3.**

```
cd fuentes && mkdir dir1 dir2 dir3
```

- 59. Crear el directorio menus bajo correo sin moverse del directorio actual.**

```
mkdir correo/menus
```

- 60. Posicionarse en el directorio HOME. Borrar los directorios que cuelgan de fuentes que acaben en un número que no sea el 1.**

```
cd $HOME && rm -r PRUEBA/dir2/fuentes/*[1]
```

- 61. Ver si existe el archivo tty2 en el directorio dev. En caso de que exista, versu fecha de creación o actualización.**

```
find /dev -type c -name "tty2" -exec ls -l \;
```

- 62. Ver los permisos que tienen los archivos que empiecen por tt del directorio /dev.**

```
ls -l /dev/tt*
```

- 63. Visualizar la lista de los archivos ordinarios que están en el directorio**

```
/usr/bin.
```

```
find /usr/bin -type d -name "*" -exec ls -l {} \;
```

- 64. Visualizar la lista de todos los directorios que cuelgan del raíz.**

```
find / -type d -name "*" -exec ls {} \;
```

- 65. Visualizar la lista de todos los ficheros que pertenezcan a root.**

```
find / -user root -type f
```

- 66. Visualizar la lista de todos los ficheros .h del directorio /usr/include.**

```
find /usr/include -type f -regex ".*.h"
```

- 67. Ejecutar todos los comandos que empiecen por ls del directorio /bin.**

```
ls /bin/ls*
```

- 68. Visualizar de qué tipo son todos y cada uno de ficheros de todo el árbol del sistema propiedad de**

un usuario conocido.

```
find /home/lorena -exec file --mime-type -0 '{}' \;
```

69. Crear el directorio uno en el directorio HOME con permiso de escritura y paso para el propietario, de lectura y paso para los usuarios de su mismo grupo y ningún permiso para el resto de usuarios.

a. `mkdir -m 660 uno && ls -ld uno`

70. Crear el directorio uno1 dentro del directorio creado en el ejercicio anterior con todos los permisos para el usuario, ninguno para los usuarios del grupo y permiso de escritura para el resto de usuarios.

```
chmod 770 uno && mkdir -pm 702 uno/uno1 && ls -ld uno/uno1
```

71. Copiar todos los ficheros propiedad de un usuario conocido que acaben en un número en el directorio menus.

```
find /home/lorena -type f -regex "[0-9]" -exec cp -r '{}' PRUEBA/dir2/correo/menus/;
```

72. Visualiza con la orden who la relación de usuarios conectados y sus terminales. Mediante la orden cat, crea un pequeño mensaje desde tu consola y redirígelo a uno de los terminales conectados.

```
who # vemos en qué terminal estamos conectados. echo "Hola usuario">/dev/tty2
```

73. Crea un archivo de tamaño 0

```
touch archivo_tamaño_cero
```

74. Visualiza el archivo /etc/motd, que contiene el "mensaje del día".

```
cat /etc/motd
```

75. Utilizando de entrada la información de los usuarios conectados al sistema, guardar, ordenadas por el campo hora, las líneas correspondientes al usuario que se desee en el archivo persona.

```
who | grep $USER | sort -k 4 > persona
```

76. Crear el directorio carpeta debajo del directorio PRUEBA. Quitarle todos los permisos de lectura. A continuación, buscar todos los directorios que cuelguen del directorio propio y guardarlos en el archivo direc.

```
mkdir -pm 333 carpeta && find ~ -type d > direc
```

77. Volver a realizar la segunda parte del ejercicio anterior, pero redireccionando los errores al fichero malos. Comprobar la información del fichero malos.

```
find ~ -type d 2> malos.
```

78. Añadir al fichero direc la lista de todos los ficheros ordinarios que cuelguen de /etc.

```
find /etc -type f >> direc
```

79. Añadir al archivo nuevalista el/los nombre/s de el/los fichero/s del directorio PRUEBA que contengan en su nombre la cadena "ai", añadiendo el posible error al fichero malos.

```
find ./ -type f -not -iname ai 1> nuevalista 2> malos | find ./ -type f -iname ai 1> nuevalista 2> malos
```

80. Sacar por pantalla únicamente el tiempo (buscar comando time) que tarda en ejecutarse el comando who.

```
time who -p %e
```

81. Sacar por pantalla un listado completo (buscar comando ps) de los procesos que está realizando el usuario root.

```
ps -U root -u root u
```

82. Crear el archivo proceso con los procesos que no tienen ningún terminal asignado.

```
ps -U root -u root u | grep -v "" | ls /dev">proceso
```

83. Añadir al fichero anterior la fecha actual y la trayectoria completa del directorio actual.

```
echo echo "$(date) - $(pwd)">>proceso
```

84. Sacar por pantalla el listado de todos los usuarios conectados ordenados por número de proceso asignado.

```
ps aux
```

85. Averiguar cuál es la actividad actual del sistema. Para ello visualice un listado completo del estado de todos los procesos que se están ejecutando en el sistema.

a. `top -d 1 -n 10`

86. Obtener un listado con los siguientes datos de los procesos de su shell actual.

a. `ps -e`

87. Mostrar cuántos usuarios tiene registrados el sistema (el registro de usuarios está en el archivo /etc/passwd)

```
cat /etc/passwd | wc -l
```

88. Mostrar cuántos usuarios tiene registrados el sistema y que utilizan el intérprete bash (debe aparecer al final de la línea /bin/bash o similar)

```
cat /etc/passwd | grep bash
```

89. Mostrar cuántos usuarios hay conectados

```
who -q
```

90. Mostrar las líneas, de un archivo de texto, empiecen por L (mayúscula o minúscula)

```
man gcc > gcc.man_page | cat gcc.man_page | sed -e 's/ //g' > file.filled | cat file.filled | grep ^[Ll]
```

91. Contar las líneas, del ejemplo anterior

```
cat file.filled | grep ^[Ll] | wc -l
```

92. Extraer los nombres de usuario (primer campo) del sistema

```
cat /etc/passwd | cut -d ':' -f 1
```

93. Extraer los nombres de usuario y el shell que utilizan (último campo)

```
awk -F: '{print $1, $7}' /etc/passwd
```

94. Cambiar la fecha de creación de un archivo ya previamente creado

```
touch good
```

```
touch -t 8810011101 good | ls -l good
```

95. Calcular la firma md5 de un archivo

```
md5sum good
```

96. Modificar la firma md5 y detectar que se ha cambiado (revisión de firma)

```
md5sum -c good.MD5 md5sum good md5sum -c good.MD5 md5sum good
```

97. Monitorear la ocupación de las particiones en los discos

```
df -lh
```

98. ¿Cuál es el proceso que más carga el procesador?

```
for x in seq 1 10; do ps -eo pid,pcpu,pmem,user,args | sort -r -k 2 | head -n 2; sleep 3; done
```

99. ¿Está corriendo el proceso bash?

```
ps -eo pid,pcpu,pmem,user,args | grep bash | ps a | grep bash
```

100. ¿Cuántos procesos que empiecen por k están corriendo?

```
ps -eo args | cut -d ' ' -f 1 | grep ^k | wc -l
```

CONCRECIÓN CURRICULAR

Este contenido, que en el temario de oposiciones corresponde a los temas del **BLOQUE 4: SISTEMAS OPERATIVOS.**, se imparte en los siguientes módulos de los diferentes niveles de los CCFF de la familia de Informática y Comunicaciones y que imparte el profesorado de Sistemas y Aplicaciones Informáticas.

Ciclo Formativo de Grado Básico o Formación Profesional Básica:

- **Informática de Oficina:** *Operaciones auxiliares para la configuración y la explotación.*
- **Informática y Comunicaciones:** *Operaciones auxiliares para la configuración y la explotación.*

Ciclo Formativo de Grado Medio:

- **Sistemas Microinformáticos y Redes:** *Sistemas Operativos Monopuesto y Sistemas Operativos en Red.*

Ciclo Formativo de Grado Superior:

- **Administración de Sistemas Informáticos en Red:** *Implantación y Sistemas Operativos y Administración de Sistemas Operativos.*
- **Desarrollo de Aplicaciones Web:** *Sistemas Informáticos.*
- **Desarrollo de Aplicaciones Multiplataforma:** *Sistemas Informáticos.*