**EJERCICIO LINUX:** (Realiza pantallazos de los pasos)

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 -dr /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

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.

$ cd /tmp

11. Verificar que el directorio actual ha cambiado.

$ pwd

12. Mostrar el día y la hora actual.

$ date +"%A %D - %r"

#ó simplemente 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 -i

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

mkdir PRUEBA

touch PRUEBA/{.f\_hidden1,.f\_hidden2,.f\_hidden3} touch PRUEBA/{file1,file2,file3}

mkdir PRUEBA/{dir1,dir2,dir3}

$ 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 PRUEBA/{ dir1,\

dir1/dir11,\ dir2,\

dir3,\ dir3/dir31,\

dir3/dir31/dir311,\ dir3/dir31/dir312}

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

$ cp /etc/motd ./PRUEBA

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 PRUEBA

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

$ cp -r /etc/rc.d dir3

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?? PRUEBA/dir3/dir31/dir311

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

$ sudo cp -r ../user\_other PRUEBA/dir1/dir11

$ cp -r ../user PRUEBA/dir1/dir11

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

$ mv PRUEBA/dir3/dir31 PRUEBA/dir2

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 PRUEBA/dir3/mensaje PRUEBA/dir3/.mensaje

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

$ rm -rf PRUEBA/dir1

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.

$ ls /dev/t???[a\*b]

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

$ find dir312 -type f -regex ".\*???q[^b$]" -exec rm -r {} \;

30. Mover el directorio dir312 debajo de dir3.

$ mv PRUEBA/dir2/dir31/dir312 PRUEBA/dir3

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

$ ln -s /home/usuario1/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 PRUEBA/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 fich1, llamarlo enlace, en los directorios dir1 y dir2.

$ ln fich1 dir1/enlace

$ ln fich1 dir2/enlace

35. Borrar el archivo fich1 y copiar enlace en dir3.

$ rm fich1

$ cp dir1/enlace dir3/

$ ln -s /home/usuario1/PRUEBA/dir2/enlace /home/usuario1/PRUEBA/dir1/enlafich1

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

$ ln -s dir2/enlace dir1/enlafich1

37. Posicionarse en dir1 y, mediante el enlace creado, copiar el archivo fichl dentro de dir311.

$ cd dir1

dir1$ cp enlafich1 ../dir2/dir31/dir311/fich1

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

dir1$ cat enlafich1

39. Borrar el fichero fich1 de dir2

PRUEBA$ rm dir2/fich1

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 dir1 dir2

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

$ chmod = dir1

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

$ chmod 751 dir2

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

$ ls -la ./dir2

45. Crear bajo dir2, un directorio llamado dir2l.

$ mkdir dir2/dir21 no se puede crear

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

$ chmod 200 dir1

$ ls -l

$ mkdir dir1/dir21

mkdir: no se puede crear el directorio «dir1/dir21»: Permiso denegado

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

$ touch dir1/{file1,file2,file3}

PRUEBA$ ls -l dir1

48. Cambiar el directorio actual al directorio dir3. Imprimir su trayectoria completa para verificar el cambio.

$ ls

dir1 dir2 dir3

$ mv dir1 dir3/

$ ls -lR

.:

./dir2:

./dir2/dir21:

./dir3:

./dir3/dir1:

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

./dir3:

50. Establecer mediante el comando umask (buscar este comando) los siguientes valores por omisión: rwxr--r-- para los directorios y rw-r--r-- para los archivos ordinarios.

umask 0033

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. Quitarle todos los permisos de lectura. Comprobarlo. Intentar borrar dicho fichero.

$ touch uno

$ chmod a-r uno

$ ls -l

$ rm uno

$ \_

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:

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 para el propietario y solo lectura para el resto.

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 carpeta1 carpeta2

$ chmod u=rwx,g=,o= carpeta1

$ chmod u=rwx,g=rx,o= carpeta2

$ ls -l

$ touch carpeta1/{fich1,fich2}

$ chmod = carpeta1/{fich1,fich2}

$ chmod o=rw carpeta1/fich1

$ ls -l carpeta1

$ touch carpeta2/{file1,file2}

$ chmod = carpeta2/{file1,file2}

$ chmod u=rw,g=rw carpeta2/file1

$ chmod u=rw,g=r carpeta2/file2

$ ls -l carpeta2

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

$su us3rlinux Contraseña:

## carpeta1 ##

# prueba de acceso us3rlinux@equipo1:/home/usuario1/PRUEBA$ cd carpeta1 bash: cd: carpeta1: Permiso denegado

# prueba de lectura us3rlinux@equipo1:/home/usuario1/PRUEBA$ ls carpeta1 ls: no se puede abrir el directorio carpeta1: Permiso denegado

## carpeta2 ##

# prueba de acceso us3rlinux@equipo1:/home/usuario1/PRUEBA$ cd carpeta2

# prueba de lectura us3rlinux@equipo1:/home/usuario1/PRUEBA/carpeta2$ ls -l total 0

-rw-rw---- 1 usuario1 usuario1 0 2009-12-08 09:41 file1

-rw-r----- 1 usuario1 usuario1 0 2009-12-08 09:41 file2

# prueba de lectura us3rlinux@equipo1:/home/usuario1/PRUEBA/carpeta2$ cat file1 us3rlinux@equipo1:/home/usuario1/PRUEBA/carpeta2$ cat file2

# prueba de escritura us3rlinux@equipo1:/home/usuario1/PRUEBA/carpeta2$ echo 'hola' > file1 us3rlinux@equipo1:/home/usuario1/PRUEBA/carpeta2$ echo 'hola' > file2 bash: file2: Permiso denegado

exit

us3rlinux@equipo1:/home/usuario1/PRUEBA$ whoami us3rlinux us3rlinux@equipo1:/home/usuario1/PRUEBA$ exit exit

usuario1@equipo1:~/PRUEBA$ whoami usuario1 usuario1@equipo1:~/PRUEBA$

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

$ pwd

/home/usuario1/PRUEBA

$ mkdir correo fuentes

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

$ cd fuentes

$ mkdir dir1 dir2

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

$ find PRUEBA/fuentes -type d -name "\*1" -exec rm -r {} \;

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

$ find PRUEBA/fuentes/\* -type d -regex ".\*[0,2,3,4,5,6,7,8,9]" -exec rm -r {} \;

$ find PRUEBA/fuentes/\* -type d -regex ".\*[^1]" -exec rm -r {} \;

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 f

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

$ ls /

$ find / -maxdepth 1 -type d

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/us3rlinux -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.

$ mkdir uno

$ chmod u=rw,g=rw,o= uno

$ ls -ld uno

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

$ chmod u=rwx,g=rwx,o= uno

$ mkdir uno/uno1

$ chmod u=rwx,g=,o=w 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/usuario1 -type f -regex ".\*[0-9]" -exec cp -r '{}' PRUEBA/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..

$ sudo -s

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

0 packages can be updated.

0 updates are security updates.

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 carpeta

$ chmod a-r 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> malo

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 `sleep 3`

$ 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`"

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

$ echo "`date +"%A %D"` - `pwd`" >>nuevalista

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

$ ps axu

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.

$ top -d .1 -n 10

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

$ cat /etc/passwd | wc -l

87. 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

88. Mostrar cuantos usuarios hay conectados

$ who -q

89. 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]

90. Contar las líneas, del ejemplo anterior

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

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

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

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

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

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

$ touch -t 9910011101 good

$ ls -l good

94. Calcular la firma md5 de un archivo

$ md5sum good

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

$ md5sum good > good.MD5

$ echo hola >> good

$ md5sum -c good.MD5

$ md5sum good

96. Monitorear la ocupación de las particiones en los discos

$ df -lh

97. ¿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

98. ¿Está corriendo el proceso bash?

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

$ ps a | grep bash

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

$ ps -eo args | cut -d ' ' -f 1 | grep ^g | wc -l

100. Realizar un script llamado '01¬hola¬mundo.sh' que muestre por pantalla "Hola mundo!"

#! /bin/bash echo "Hola mundo!"