**1. Línea de comandos**

**1.1. Utilizando el manual del sistema:**

**Esta parte propone unas tareas con el manual del sistema. Es un recurso incluido en la mayoría de sistemas**

**Unix/Linux y es muy útil en situaciones en las que no exista acceso a internet u otra fuente de información o**

**documentación.**

**1) Entrar en la consola de la máquina virtual de Linux.**

xd

**2) Usar man man para ver la ayuda del manual.**

$ man man

**3) Pulsar ‘h’ para mostrar las teclas y comandos para moverse por las páginas de man.**

xd

**4) Salir del manual.**

q

**5) Entrar de nuevo en el manual y:**

**a. Moverse al comienzo y final de la página.**

g → Comienzo

G → Final

**b. Moverse línea a línea arriba y abajo por el documento.**

↑ → Arriba

↓ → Abajo

**c. Buscar la palabra “word” en el documento. Navegar por todas las ocurrencias de la palabra.**

/word

n → Siguiente

N → Anterior

**d. Saltar al final del manual y después al comienzo de nuevo.**

g → Comienzo

G → Final

**1.2. Gestión y manipulación de archivos**

**Esta parte está enfocada a explorar diferentes comandos del Shell y ver su efecto.**

**1) Crear una carpeta llamada AS en vuestro directorio raíz de usuario.**

$ mkdir AS

**2) Entrar dentro de la carpeta y comprobar que el directorio coincide con el contenido de la variable de entorno PWD.**

ibaio@instantzia-sa:~$ cd AS

ibaio@instantzia-sa:~/AS$ pwd

/home/ibaio/AS

ibaio@instantzia-sa:~/AS$ echo "$PWD"

/home/ibaio/AS

**3) Instalar cal con el comando apt install ncal. Utilizar cal para mostrar un calendario y redirigir la salida a un fichero de texto. Comprobar que ese fichero se crea correctamente y que su contenido es el esperado.**

$ sudo apt install ncal

ibaio@instantzia-sa:~/AS$ ncal > calendario.txt

ncal: setlocale: No such file or directory

ibaio@instantzia-sa:~/AS$ cat calendario.txt

October 2025

Su 5 12 19 26

Mo 6 13 20 27

Tu 7 14 21 28

We 1 8 15 22 29

Th 2 9 16 23 30

Fr 3 10 17 24 31

Sa 4 11 18 25

**4) Copiar el fichero recién creado al directorio raíz del usuario.**

~/AS$ cp calendario.txt "$HOME"

**5) Moverse al directorio raíz del usuario y listar en formato extendido (parámetro -l) los directorios y archivos presentes. Redirigir esa información a un fichero.**

$ ls -la > listado\_home.txt

ibaio@instantzia-sa:~$ head -n 30 listado\_home.txt

**6) Listar los 5 ficheros más nuevos de la carpeta /etc (que no sean carpetas).**

$ find /etc -maxdepth 1 -type f -printf '%T@ %p\n' 2>/dev/null | sort -nr | head -n 5 | cut -d' ' -f2-

**7) Cambiar los permisos del fichero creado con información del calendario para que sólo el usuario propietario tenga capacidad de leer y escribir.**

$ chmod 600 calendario.txt

ibaio@instantzia-sa:~$ ls -l calendario.txt

**8) Cambiar permisos para evitar que ningún otro usuario pueda acceder nuestro directorio raíz.**

$ chmod go-rwx "$HOME"

ibaio@instantzia-sa:~$ ls -ld "$HOME"

**9) Comprobar cuántos usuarios hay en el sistema y cual es el nuestro Shell de inicio.**

# número de entradas en /etc/passwd (cuentas del sistema/usuarios):

wc -l /etc/passwd

# o solo nombres:

awk -F: '{print $1}' /etc/passwd | wc -l

# shell de inicio real del usuario:

getent passwd "$USER" | cut -d: -f7

# (también puedes ver $SHELL)

echo "$SHELL"

**10) Comprobar cuándo y desde donde accediste la última vez al sistema.**

$ last -n 1 ibaio

**11) Comprimir en un archivo .tar.gz los contenidos del directorio $HOME. Descomprimirlos /tmp y comprobar que se ha hecho correctamente.**

$ tar -czf /tmp/home\_contents.tar.gz -C "$HOME" .

$ mkdir /tmp/home\_extracted

$ tar -xzf /tmp/home\_contents.tar.gz -C /tmp/home\_extracted

$ ls -la /tmp/home\_extracted | head -n 50

**12) Como usuario “root”, buscar todos los archivos que sean propiedad de tu usuario en el sistema (desde el directorio raíz /) y listarlos en forma extendida (con toda la información).**

$ sudo find / -user ibaio -ls 2>/dev/null > /tmp/archivos\_de\_mi\_usuario.txt

ibaio@instantzia-sa:~$ sudo head -n 50 /tmp/archivos\_de\_mi\_usuario.txt

**13) Como usuario “root”, mostrar las últimas 30 líneas de /var/log/syslog.**

$ sudo tail -n 30 /var/log/syslog

**2. Shell Scripting**

**Esta parte del laboratorio propone crear varios scripts utilizando los comandos aprendidos en la parte anterior.**

**1) Crear un script llamado lsdirs.sh que muestre los directorios contenidos en el directorio actual.**

$ vim lsdirs.sh

#!/bin/bash

shopt -s nullglob

dirs=(\*/)

if [ ${#dirs[@]} -eq 0 ]; then

echo "No hay directorios en $(pwd)"

exit 0

fi

for d in "${dirs[@]}"; do

echo "${d%/}"

done

$ chmod +x lsdirs.sh

$ ./lsdirs.sh

**2) Crear un script llamado see.sh que reciba un nombre de fichero/directorio como parámetro. Si el nombre corresponde a un fichero, el script muestra el contenido del fichero con more, sino muestra el contenido del directorio con ls.**

$ vim see.sh

#!/bin/bash

if [ -z "$1" ]; then

echo "Uso: $0 fichero\_o\_directorio"

exit 1

fi

if [ -f "$1" ]; then

more "$1"

elif [ -d "$1" ]; then

ls -la "$1"

else

echo "'$1' no existe o no es fichero/directorio"

fi

$ chmod +x see.sh

$ ./see.sh calendario.txt

$ ./see.sh AS

**3) Crear un script que pida al usuario que teclee una palabra y escriba por pantalla el número de caracteres de esa palabra.**

$ vim contLetr.sh

#!/bin/bash

read -p "Escribe una palabra: " palabra

echo "Número de caracteres: ${#palabra}"

$ chmod +x contLetr.sh

$ ./contLetr.sh

**4) Crear un script que pida al usuario que teclee una palabra y compruebe si es un comando del sistema o no.**

$ vim es\_comando.sh

#!/bin/bash

read -p "Teclea un nombre para comprobar si es comando: " cmd

if command -v "$cmd" >/dev/null 2>&1; then

echo "'$cmd' es un comando del sistema: $(command -v "$cmd")"

else

echo "'$cmd' NO es un comando del sistema"

fi

$ chmod +x es\_comando.sh

ibaio@instantzia-sa:~/lab1$ ./es\_comando.sh

**5) Crear un script que cree una carpeta llamada cosas y después cree 100 ficheros vacíos llamados fich<numero>.txt dentro de ella, donde <numero> es un número entre 0 y 99.**

mkdir -p cosas

for i in $(seq 0 99); do

touch "cosas/fich${i}.txt"

done

echo “Creados 100 ficheros en /cosas"

$ ./contLetr.sh

**6) Extender el script anterior para que cada fichero contenga la N-sima línea del manual de ls (man ls). El fichero fich0.txt tendrá la línea 0 del manual, fich1.txt tendrá la línea 1 del manual,…**

#!/bin/bash

mkdir -p cosas

man ls > /tmp/man\_ls.txt

for i in $(seq 0 99); do

line\_num=$((i+1)) # tratamos "línea 0" como la primera línea del fichero

sed -n "${line\_num}p" /tmp/man\_ls.txt > "cosas/fich${i}.txt"

done

rm -f /tmp/man\_ls.txt

echo "Ficheros creados y rellenados en ./cosas"

**7) Crear un script que modifique la extensión de todos los ficheros .txt de un directorio a .t.**

#!/bin/bash

shopt -s nullglob

for f in \*.txt; do

mv -- "$f" "${f%.txt}.t"

done

echo "Conversión .txt -> .t completada."

./txt\_to\_t

**8) Crear un script borra.sh que reciba un número indefinido de parámetros (de 0 a 9) y borre el fichero correspondiente a la suma del valor de los parámetros que reciba. Por ejemplo, borra.sh 1 4 5 9, borraría el fichero fich19.txt (1 + 4 + 5 + 9 = 19).**

#!/bin/bash

if [ $# -eq 0 ]; then

echo "Uso: $0 n1 n2 ..."

exit 1

fi

sum=0

for p in "$@"; do

if ! [[ "$p" =~ ^[0-9]+$ ]]; then

echo "Parámetro inválido (no numérico): $p" >&2

exit 2

fi

sum=$((sum + p))

done

target="fich${sum}.txt"

if [ -e "$target" ]; then

rm -v "$target"

else

echo "El fichero $target no existe."

fi

./borra.sh 1 2 3

**9) Crear un script orden.sh que muestre el contenido del fichero /etc/passwd ordenado por nombre de usuario, UID o GID. El script recibirá un parámetro que indique cuál de los 3 utilizar como clave de ordenación.**

#!/bin/bash

case "$1" in

user)

sort -t: -k1,1 /etc/passwd

;;

uid)

sort -t: -n -k3,3 /etc/passwd

;;

gid)

sort -t: -n -k4,4 /etc/passwd

;;

\*)

echo "Uso: $0 {user|uid|gid}"

exit 1

;;

esac

./orden.sh user

./orden.sh uid

./orden.sh gid

**10) Imagina que quieres mandar un e-mail con el mismo cuerpo a varios destinatarios, pero personalizando la primera línea con el nombre de cada uno. Crear un fichero cuerpo.txt con el texto del cuerpo del e-mail y la palabra NOMBRE en cada lugar donde querrías poner el nombre, y un fichero nombres.txt con varios nombres personales. Crear un script que genere varios ficheros con el cuerpo del e-mail personalizado para cada destinatario.**

$ cat > cuerpo.txt <<'EOF'

Hola NOMBRE,

Esto es el cuerpo del correo.

Saludos.

EOF

$ cat > nombres.txt <<'EOF'

Ana

Carlos

María

EOF

$ cat > personaliza\_correos.sh <<'EOF'

#!/bin/bash

if [ ! -f cuerpo.txt ]; then

echo "Falta cuerpo.txt" >&2

exit 1

fi

if [ ! -f nombres.txt ]; then

echo "Falta nombres.txt" >&2

exit 1

fi

mkdir -p correos\_personalizados

while IFS= read -r nombre; do

# reemplazar palabra NOMBRE por el nombre (escape para nombres con caracteres especiales)

sed "s/NOMBRE/${nombre}/g" cuerpo.txt > "correos\_personalizados/correo\_${nombre}.txt"

done < nombres.txt

echo "Generados correos en ./correos\_personalizados"

EOF

$ chmod +x personaliza\_correos.sh

./personaliza\_correos.sh