4.36

a) **Ejecuta la orden** sh:

* Al ejecutar sh, la terminal lanza una nueva instancia del intérprete de comandos Bourne shell. Esto te lleva a un nuevo nivel de shell donde puedes ejecutar otros comandos.

b) **Ejecuta el comando** exit:

* El comando exit termina la sesión actual del shell. Si ejecutas exit después de sh, cerrarás la instancia de Bourne shell en la que estabas y volverás al shell original de la terminal.

c) **Ejecuta** dash:

* Al ejecutar dash, la terminal lanza el intérprete de comandos Debian Almquist Shell. Similar a sh, esto te lleva a un nuevo nivel de shell donde puedes ejecutar otros comandos.

d) **Ejecuta el comando** exit:

* El comando exit termina la sesión actual del shell. Si ejecutas exit después de dash, cerrarás la instancia de Debian Almquist Shell en la que estabas y volverás al shell original de la terminal.

4.37

a) **CTRL+ALT+F3**:

* Te lleva a la tercera terminal virtual (TTY3). Aquí verás una interfaz de línea de comandos y puedes iniciar sesión con tus credenciales.

b) **CTRL+ALT+F5**:

* Te lleva a la quinta terminal virtual (TTY5). Similar a TTY3, tendrás una interfaz de línea de comandos para iniciar sesión.

c) **CTRL+ALT+F2**:

* Te lleva a la segunda terminal virtual (TTY2), donde también verás una interfaz de línea de comandos.

d) **CTRL+ALT+F4**:

* Te lleva a la cuarta terminal virtual (TTY4), otra vez mostrando una interfaz de línea de comandos.

e) **CTRL+ALT+F1**:

* En muchas distribuciones de Linux, esto te regresa al entorno gráfico (TTY1), donde se encuentra tu sesión de escritorio actual.

4.38

a) help -d echo:

* La opción -d muestra una breve descripción de la orden echo. En este caso, devuelve una descripción corta explicando qué hace el comando echo.

b) help -m echo:

* La opción -m muestra la documentación completa del comando echo. Esto incluye detalles extensivos sobre cómo se utiliza echo, las opciones disponibles y ejemplos de uso.

c) help -s echo:

* La opción -s muestra un resumen conciso sobre el uso del comando echo. Proporciona una visión general del comando y sus opciones principales sin entrar en detalles extensivos.

4.39

a) **Tecla Av Pág (Page Down) o barra espaciadora**:

* Ambas teclas te permiten desplazarte hacia abajo en la página de manual, mostrando el siguiente segmento de contenido.

b) **Tecla Re Pág (Page Up)**:

* Esta tecla te permite desplazarte hacia arriba en la página de manual, mostrando el segmento anterior de contenido.

c) **Teclas de flecha hacia arriba y flecha hacia abajo**:

* La flecha hacia arriba te desplaza una línea hacia arriba en la página de manual.
* La flecha hacia abajo te desplaza una línea hacia abajo en la página de manual.

d) **Tecla q**:

* Pulsar q te permite salir de la página de manual y regresar a la terminal.

4.40

a) **Tecla flecha hacia abajo y flecha hacia arriba**:

* La flecha hacia abajo desplaza el contenido una línea hacia abajo.
* La flecha hacia arriba desplaza el contenido una línea hacia arriba.

b) **Barra espaciadora o Av Pág (Page Down)**:

* Ambas teclas desplazan el contenido una página hacia abajo, mostrando el siguiente segmento de información.

c) **Tecla q**:

* Pulsar q cierra la página de información y te regresa a la terminal.

4.41

a)

B)

c)



d)



4.42

a)



b)



c)



d)



e)



4.43

a)



b)



c)



d)





e)





4.44

a) ls texto[X-Z]:

* Lista los archivos en el directorio actual cuyo nombre comienza con "texto" y termina con una letra que se encuentra entre X y Z (mayúsculas). Por ejemplo, puede listar textoX, textoY y textoZ.

b) ls [3-5]??f\*:

* Lista los archivos cuyos nombres comienzan con un número entre 3 y 5, seguidos de exactamente dos caracteres (pueden ser cualquier carácter), seguidos de la letra f y luego cualquier otro número de caracteres. Por ejemplo, 3abf, 4xxfabc, 5yof serían coincidencias.

c) ls [dD]ocumentos:

* Lista los archivos o directorios llamados documentos o Documentos en el directorio actual, porque [dD] coincide con d o D.

d) ls fichero[^1234]:

* Lista los archivos cuyo nombre comienza con fichero y tienen un carácter a continuación que no sea 1, 2, 3, o 4. Por ejemplo, ficheroA, ficherob serían coincidencias, pero fichero1 no.

e) ls c[ao]sa:

* Lista los archivos llamados casa o cosa en el directorio actual, porque [ao] coincide con a o o.

f) ls [a-z]\*:

* Lista todos los archivos cuyos nombres comienzan con una letra minúscula (de a a z). Cualquier archivo que comience con una letra mayúscula, número o carácter especial no será listado.

g) ls [a-zA-Z]\*:

* Lista todos

4.45

a) **Cualquier archivo o directorio que contenga las letras "a" y/o "e"**:

ls \*[ae]\*

b) **Cualquier archivo o directorio que no comience con una vocal**:

ls [!aeiouAEIOU]\*

c) **Cualquier archivo o directorio que empiece por una mayúscula**:

ls [A-Z]\*

d) **Cualquier archivo o directorio que empiece por una letra minúscula, siga con los caracteres “eso” y tenga como extensión “odt”**:

ls [a-z]eso.odt

e) **Cualquier fichero o directorio cuyo nombre tenga 3 caracteres de longitud y que empiece por un número**:

ls [0-9][a-zA-Z0-9][a-zA-Z0-9]

4.46

a) **Realiza un listado recursivo de tu carpeta personal**:

ls -R ~

Esto listará todos los archivos y directorios dentro de tu carpeta personal y sus subdirectorios de manera recursiva.

b) **Haz que se muestren los ficheros y directorios ocultos de tu carpeta personal**:

ls -a ~

La opción -a muestra todos los archivos, incluyendo los ocultos (cuyo nombre empieza con un punto .).

c) **¿Cuál es el tamaño del fichero** .bash\_history**?**:

du -h ~/.bash\_history

Esto muestra el tamaño del archivo .bash\_history en un formato legible para los humanos.

d) **¿Hay un directorio de nombre “Descargas”?**:

if [ -d ~/Descargas ]; then echo "El directorio 'Descargas' existe"; else echo "El directorio 'Descargas' no existe"; fi

Este comando verificará si el directorio Descargas existe en tu carpeta personal.

e) **¿Hay un directorio de nombre “.bashrc”?**:

if [ -f ~/.bashrc ]; then echo "El archivo '.bashrc' existe"; else echo "El archivo '.bashrc' no existe"; fi

Este comando verificará si el archivo .bashrc existe en tu carpeta personal.

4.47

Abre una terminal y ejecuta las siguientes órdenes:

1. **Crear el directorio principal** ventas:

mkdir -p ~/ventas/nacionales ~/ventas/internacionales/europa

1. **Crear los archivos en** ventas:

touch ~/ventas/nacionales/primersemestre.txt ~/ventas/nacionales/segundosemestre.txt

touch ~/ventas/internacionales/primertrimestre.txt ~/ventas/internacionales/segundotrimestre.txt

touch ~/ventas/internacionales/europa/enero.txt ~/ventas/internacionales/europa/febrero.txt

1. **Crear el directorio principal** compras:

mkdir -p ~/compras/proveedores

1. **Crear los archivos en** compras:

touch ~/compras/proveedores/nacionales.txt ~/compras/proveedores/internacionales.txt

1. **Crear el directorio principal** nominas:

mkdir -p ~/nominas/empleados

1. **Crear los archivos en** nominas:

touch ~/nominas/empleados/datos.txt

4.48

a) **Copia el fichero** datos.txt **en el directorio** proveedores:

cp ../nominas/empleados/datos.txt ../compras/proveedores/

b) **Copia el contenido del directorio** nacionales **en el directorio** internacionales:

cp -r nacionales/\* internacionales/

c) **Copia el fichero** enero.txt **en el directorio** europa **con el nombre** enero2017.txt:

cp internacionales/europa/enero.txt internacionales/europa/enero2017.txt

d) **Copia todos los ficheros o directorios del directorio** europa **que empiecen por la letra** e **al directorio** proveedores:

cp internacionales/europa/e\* ../compras/proveedores/

e) **Copia el directorio** ventas **con el nombre** ventas2:

cp -r ../ventas ../ventas2

4.49

a) **Borrar el contenido del directorio** ventas2**, pero no el directorio**:

rm -r ../ventas2/\*

b) **Borrar en el directorio** europa **los ficheros y directorios que empiecen por “e”**:

rm -r internacionales/europa/e\*

c) **Borrar el directorio** ventas2:

rmdir ../ventas2

4.50

a) **Renombrar el fichero** datos.txt **como** datos:

mv ../nominas/empleados/datos.txt ../nominas/empleados/datos

b) **Mover todos los ficheros de** ventas/internacionales **a** ventas:

mv internacionales/\* .

c) **Renombrar el directorio** ventas **como** ventasempresa:

cd ..  
mv ventas ventasempresa

4.51

a) ls -l . >> listado:

* Esta orden añade (sin sobrescribir) la salida del comando ls -l . (un listado detallado del contenido del directorio actual) al final del archivo listado. Si el archivo listado no existe, se creará.

b) cat < archivo1 > archivo2:

* Esta orden redirige el contenido del archivo archivo1 y lo escribe en el archivo archivo2. Si archivo2 existe, su contenido será sobrescrito. Si archivo2 no existe, se creará.

c) cat notas faltas > alumnos:

* Esta orden concatena el contenido de los archivos notas y faltas, y escribe el resultado en el archivo alumnos. Si alumnos existe, su contenido será sobrescrito. Si alumnos no existe, se creará.

d) cat actividades >> alumnos:

* Esta orden añade (sin sobrescribir) el contenido del archivo actividades al final del archivo alumnos. Si alumnos no existe, se creará.

4.52

a) **Guardar el listado de los ficheros y directorios en** /home/usuario **en el fichero** listado:

ls /home/usuario > listado

b) **Concatenar los ficheros** uno **y** dos **en el fichero** juntos:

cat uno dos > juntos

c) **Sobrescribir el fichero** copia **con el contenido del fichero** hoy:

cat hoy > copia

d) **Añadir al final del fichero** semana **el contenido del fichero** jueves:

cat jueves >> semana

4.53

a) sort frutas:

* Ordena el contenido del fichero en orden alfabético ascendente basado en la primera columna.
* Salida:

1 Pomelo  
2 Pera  
2 Pera  
3 Manzana  
4 Cereza  
6 Kiwi

b) sort -r frutas:

* Ordena el contenido del fichero en orden alfabético descendente basado en la primera columna.
* Salida:

6 Kiwi  
4 Cereza  
3 Manzana  
2 Pera  
2 Pera  
1 Pomelo

c) sort -u frutas:

* Ordena el contenido del fichero en orden alfabético ascendente y elimina duplicados.
* Salida:

1 Pomelo  
2 Pera  
3 Manzana  
4 Cereza  
6 Kiwi

d) sort -k 2 frutas:

* Ordena el contenido del fichero basado en la segunda columna (el nombre de la fruta) en orden alfabético ascendente.
* Salida:

4 Cereza  
6 Kiwi  
3 Manzana  
2 Pera  
2 Pera  
1 Pomelo

e) sort -nr frutas:

* Ordena el contenido del fichero basado en la primera columna en orden numérico descendente.
* Salida:

6 Kiwi  
4 Cereza  
3 Manzana  
2 Pera  
2 Pera  
1 Pomelo

Estas órdenes permiten ordenar el contenido del fichero frutas de diferentes maneras utilizando opciones específicas del comando sort

4.54

a) tar -cvf notas.tar notas1 eval:

* Crea un archivo tar (notas.tar) que incluye los directorios y/o archivos notas1 y eval.

b) tar -tf notas.tar:

* Lista el contenido del archivo tar notas.tar.

c) tar -rvf notas.tar notasjunio:

* Añade el archivo o directorio notasjunio al archivo tar existente notas.tar.

d) tar -xvf notas.tar:

* Extrae el contenido del archivo tar notas.tar.

e) gzip -v notas.tar:

* Comprime el archivo notas.tar utilizando gzip, creando notas.tar.gz.

f) tar tfz notas.tar.gz:

* Lista el contenido del archivo comprimido notas.tar.gz.

g) gzip -d notas.tar.gz:

* Descomprime el archivo notas.tar.gz, devolviéndolo a su estado original notas.tar.

4.55

a) **Empaquetar el directorio** ventasempresa:

tar -cvf ventasempresa.tar ventasempresa

b) **Añadir al archivo** ventasempresa.tar **el directorio** nominas:

tar -rvf ventasempresa.tar nominas

c) **Visualizar el contenido del archivo** ventasempresa.tar:

tar -tf ventasempresa.tar

d) **Comprimir el archivo** ventasempresa.tar:

gzip -v ventasempresa.tar

e) **¿Cuál ha sido el factor de compresión?**:

* Después de ejecutar el comando anterior, la salida del comando gzip indicará el tamaño original y el tamaño comprimido del archivo, lo que te permitirá calcular el factor de compresión. Por ejemplo:

ventasempresa.tar: 70.5% -- replaced with ventasempresa.tar.gz

Aquí, el factor de compresión es el 70.5%.

f) **Listar el contenido del archivo comprimido** ventasempresa.tar.gz:

tar -tfz ventasempresa.tar.gz

g) **Descomprimir el archivo**:

gzip -d ventasempresa.tar.gz

4.56

1. **Crear un fichero vacío de nombre** enblanco:

touch enblanco

a) **Ejecuta** file enblanco:

file enblanco

b) **Abre el fichero** enblanco **y escribe algo dentro. Luego ejecuta** file enblanco:

* Abre el fichero con un editor de texto y escribe algo. Por ejemplo:

echo "Este es un fichero de prueba" > enblanco

* Luego ejecuta:

file enblanco

c) **Comprime el fichero** enblanco**. Luego ejecuta** file enblanco.gz:

* Comprime el fichero:

gzip enblanco

* Luego ejecuta:

file enblanco.gz

d) **Ejecuta** file -z enblanco.gz:

file -z enblanco.gz

e) **Ejecuta** file \*:

file \*

4.57

Estos son los resultados esperados de los comandos find que has mencionado. Ejecuta estos comandos en una terminal de Linux o macOS para verlos en tu sistema:

1. *a) find / –name “[0-9]*” 2>/dev/null\*

find / –name "[0-9]\*" 2>/dev/null

Este comando buscará archivos y directorios cuyo nombre contenga cualquier carácter numérico en todo el sistema, suprimiendo los mensajes de error.

1. **b) find –size +100k**

find . -size +100k

Este comando buscará archivos en el directorio actual y sus subdirectorios que tengan un tamaño superior a 100 kilobytes.

1. **c) find / –mtime -4 2>/dev/null**

find / -mtime -4 2>/dev/null

Este comando buscará archivos y directorios en todo el sistema que hayan sido modificados en los últimos 4 días, suprimiendo los mensajes de error.

1. **d) find . –type f –name archivo1**

find . -type f -name archivo1

Este comando buscará archivos regulares llamados "archivo1" en el directorio actual y sus subdirectorios.

1. *e) find / –user juancarlos –not –name “*do” 2> /dev/null\*

find / -user juancarlos -not -name "\*do" 2>/dev/null

Este comando buscará archivos y directorios en todo el sistema propiedad del usuario "juancarlos" y cuyos nombres no terminan en "do", suprimiendo los mensajes de error.

1. *f) find . -type f –name "*.odt" -o –name ".sh"\*\*

find . -type f -name "\*.odt" -o -name "\*.sh"

Este comando buscará archivos en el directorio actual y sus subdirectorios que tengan la extensión ".odt" o ".sh".

1. *g) find . -maxdepth 1 -type f -name "*.odt"\*

find . -maxdepth 1 -type f -name "\*.odt"

Este comando buscará archivos con la extensión ".odt" en el directorio actual sin buscar en subdirectorios (profundidad máxima de 1).

1. **h) find . -type f -maxdepth 1 -empty -ok rm {} \;**

find . -type f -maxdepth 1 -empty -ok rm {} \;

Este comando buscará archivos vacíos en el directorio actual sin buscar en subdirectorios (profundidad máxima de 1) y, si se encuentran, pedirá confirmación antes de eliminarlos.

4.58

* **a) Buscar en el directorio actual y subdirectorios ficheros que se llamen "notas" sin importar si está escrito en mayúsculas o minúsculas:**

find . -iname "notas"

* **b) Buscar dentro de tu carpeta personal ficheros de más de 1 KB y menos de 1 MB:**

find ~ -size +1k -size -1M

* **c) Localizar en /tmp los ficheros modificados en los últimos dos días:**

find /tmp -mtime -2

* **d) Localizar en /tmp los ficheros creados hace más de cinco días:**

find /tmp -ctime +5

* **e) Buscar ficheros y directorios del usuario root que se llamen "snap":**

find / -user root -name "snap" 2>/dev/null

* **f) Buscar los directorios vacíos en /home:**

find /home -type d -empty

* **g) Borrar los ficheros de extensión “.odt” en tu directorio personal:**

find ~ -type f -name "\*.odt" -delete

* **h) Borrar en el directorio actual los directorios cuyo nombre empiece por “d”:**

find . -maxdepth 1 -type d -name "d\*" -exec rm -r {} \;

4.59

* **a) Todas las líneas que contienen la cadena “frase”:**

grep "frase" archivo1

* **b) El número de líneas que contienen la cadena “frase”:**

grep -c "frase" archivo1

* **c) Las líneas que empiezan por “Segunda”:**

grep "^Segunda" archivo1

* **d) Las líneas que empiezan por “Segunda” o “segunda”:**

grep -i "^Segunda" archivo1

* **e) Las líneas que empiezan con letras de la “M” a la “V”:**

grep "^[M-V]" archivo1

* **f) El número de línea donde se encuentra la palabra “Tercera”:**

grep -n "Tercera" archivo1

* **g) Las líneas que terminan por “e”:**

grep "e$" archivo1

* **h) Las líneas que contienen alguna vocal:**

grep -i "[aeiou]" archivo1

* **i) Las líneas que contienen algún número entre 2 y 5:**

grep "[2-5]" archivo1

* **j) Las líneas que comienzan por un número:**

grep "^[0-9]" archivo1

4.60

* **a) Crea un enlace simbólico de nombre “logs” al directorio “/var/log”:**

ln -s /var/log logs

* **b) Ejecuta “cd logs”, ¿te ha llevado al directorio /var/log?**

cd logs

Sí, esto te debería llevar al directorio /var/log. Puedes comprobarlo ejecutando:

pwd

Debería mostrar /var/log.

* **c) Crea un enlace simbólico de nombre “equipo” al fichero “/etc/hostname”:**

ln -s /etc/hostname equipo

* **d) Ejecuta “cat equipo”, ¿has visualizado el nombre del equipo?**

cat equipo

Esto debería mostrar el contenido del fichero /etc/hostname, que es el nombre del equipo.

* **e) Elimina los enlaces simbólicos “logs” y “equipo”:**

rm logs equipo

4.61

* **a) Muestra el primer y el tercer campo del fichero /etc/passwd:**

awk -F: '{print $1, $3}' /etc/passwd

* **b) Visualiza los cinco primeros caracteres del fichero /etc/passwd:**

head -c 5 /etc/passwd

* **c) Muestra el número de palabras del fichero .bashrc (está en tu carpeta personal):**

wc -w ~/.bashrc

* **d) Visualiza el tipo de contenido del fichero .bashrc:**

file ~/.bashrc

* **e) Muestra el tamaño en bytes del fichero .bashrc:**

stat --format=%s ~/.bashrc

* **f) La opción –b del comando du muestra el tamaño del fichero en bytes. Utilízalo para obtener el tamaño del fichero .bashrc:**

du -b ~/.bashrc

4.62

**a)** chmod u=rwx fich1: Este comando establece los permisos del usuario (propietario) del archivo fich1 para que tenga permisos de lectura (r), escritura (w) y ejecución (x). No afecta a los permisos del grupo ni de otros.

* Usuario (u): rwx (lectura, escritura y ejecución)
* Grupo (g): Sin cambios
* Otros (o): Sin cambios

**b)** chmod g+rx fich2: Este comando agrega permisos de lectura (r) y ejecución (x) para el grupo en el archivo fich2. No afecta a los permisos del usuario ni de otros.

* Usuario (u): Sin cambios
* Grupo (g): +r+x (agrega lectura y ejecución)
* Otros (o): Sin cambios

**c)** chmod g-wx fich3: Este comando elimina los permisos de escritura (w) y ejecución (x) para el grupo en el archivo fich3. No afecta a los permisos del usuario ni de otros.

* Usuario (u): Sin cambios
* Grupo (g): -w-x (elimina escritura y ejecución)
* Otros (o): Sin cambios

**d)** chmod 754 fich4: Este comando establece los permisos del archivo fich4 usando una notación numérica.

* Usuario (u): rwx (lectura, escritura y ejecución)
* Grupo (g): r-x (lectura y ejecución)
* Otros (o): r-- (solo lectura)

**e)** chmod 400 fich5: Este comando establece los permisos del archivo fich5 usando una notación numérica.

* Usuario (u): r-- (solo lectura)
* Grupo (g): --- (sin permisos)
* Otros (o): --- (sin permisos)

4.63

* **a) Sobre el fichero “archivo1” para que el grupo pueda leerlo y ejecutarlo, pero no modificarlo:**

chmod g=rx archivo1

* **b) Sobre el fichero “archivo2” para que el usuario pueda leerlo y ejecutarlo y el grupo y el resto sólo leerlo:**

chmod u=rx,g=r,o=r archivo2

* **c) Sobre el fichero “archivo3” para que el grupo no pueda modificarlo, ni ejecutarlo:**

chmod g-wx archivo3

* **d) Sobre el fichero “archivo4” para que el usuario propietario tenga todos los permisos y el grupo y el resto sólo de lectura:**

chmod 744

4.64

* **a) Ejecuta la orden** umask

umask

El valor que verás en la salida será el valor actual de la máscara de umask en tu sistema. Por ejemplo, podría ser 0022 o algo similar.

* **b) Aplica una máscara para que los nuevos permisos de los ficheros que se creen sean 640**

umask 0137

La máscara 0137 asegura que los permisos resultantes para los nuevos archivos sean 640.

* **c) Crea un fichero y comprueba la nueva máscara**

touch nuevo\_fichero  
ls -l nuevo\_fichero

Verás que los permisos del archivo nuevo\_fichero son -rw-r-----, que corresponde a 640.

* **d) Establece la máscara de nuevo a “0002”**

umask 0002

4.65

* **a) Localiza en el fichero /etc/passwd la línea correspondiente al usuario que creaste durante la instalación y explica cada uno de los campos.**
* Abre el fichero /etc/passwd:

cat /etc/passwd

* Localiza la línea correspondiente a tu usuario (por ejemplo, si tu usuario es "juan"):

juan:x:1000:1000:Juan User,,,:/home/juan:/bin/bash

* Explicación de los campos (separados por dos puntos :):
  1. **Nombre de usuario**: juan
  2. **Contraseña (x indica que está encriptada)**: x
  3. **UID (User ID)**: 1000
  4. **GID (Group ID)**: 1000
  5. **GECOS (nombre completo y otra información)**: Juan User,,,
  6. **Directorio home**: /home/juan
  7. **Shell predeterminada**: /bin/bash
* **b) Localiza en el fichero /etc/passwd la línea correspondiente al usuario root y explica cada uno de los campos.**
* Localiza la línea correspondiente al usuario root:

root:x:0:0:root:/root:/bin/bash

* Explicación de los campos (separados por dos puntos :):
  1. **Nombre de usuario**: root
  2. **Contraseña (x indica que está encriptada)**: x
  3. **UID (User ID)**: 0
  4. **GID (Group ID)**: 0
  5. **GECOS (nombre completo y otra información)**: root
  6. **Directorio home**: /root
  7. **Shell predeterminada**: /bin/bash

4.66

* **a) Crea un usuario de nombre “pedro” y su directorio personal:**

sudo useradd -m pedro

* **b) Lista el directorio** /home **y comprueba que aparece su carpeta personal:**

ls /home

Deberías ver un directorio llamado pedro en la lista.

* **c) Comprueba en el fichero** /etc/passwd **que aparece una línea para el usuario “pedro”:**

grep "pedro" /etc/passwd

Deberías ver una línea similar a esta:

pedro:x:1001:1001::/home/pedro:/bin/bash

* **d) Establece la contraseña para el usuario “pedro”:**

sudo passwd pedro

Se te pedirá que ingreses y confirmes la nueva contraseña.

* **e) Cambia el nombre del usuario “pedro” por “pedrito”:**

sudo usermod -l pedrito pedro  
sudo usermod -d /home/pedrito -m pedrito

* **f) Comprueba en el fichero** /etc/passwd **el cambio:**

grep "pedrito" /etc/passwd

Deberías ver una línea similar a esta:

pedrito:x:1001:1001::/home/pedrito:/bin/bash

* **g) Cambia el UID del usuario “pedrito” por 1500:**

sudo usermod -u 1500 pedrito

* **h) Comprueba en el fichero** /etc/passwd **el cambio:**

grep "pedrito" /etc/passwd

Deberías ver una línea similar a esta:

pedrito:x:1500:1001::/home/pedrito:/bin/bash

* **i) Elimina el usuario “pedrito” y su directorio personal:**

sudo userdel -r pedrito

* **j) Lista el directorio** /home**, ¿aparece el directorio “pedrito”?**

ls /home

4.67

* **a) Localiza en el fichero** /etc/group **la línea que corresponde al grupo principal del usuario que creaste durante la instalación y explica su significado:**
* Abre el fichero /etc/group y busca el grupo principal del usuario (por ejemplo, si tu usuario es "juan"):

grep "juan" /etc/group

* Supongamos que encuentras una línea como esta:

juan:x:1001:

* Explicación de los campos (separados por dos puntos :):
  1. **Nombre del grupo**: juan
  2. **Contraseña (x indica que está encriptada)**: x
  3. **GID (Group ID)**: 1001
  4. **Lista de usuarios miembros**: En este caso, está vacío, lo que significa que este grupo no tiene usuarios adicionales aparte del propio "juan".
* **b) Localiza en el fichero** /etc/group **todos los grupos a los que pertenece el usuario que creaste durante la instalación:**
* Para encontrar todos los grupos a los que pertenece el usuario (por ejemplo, "juan"):

grep "juan" /etc/group

* Supongamos que encuentras varias líneas como estas:

juan:x:1001:  
sudo:x:27:juan  
docker:x:999:juan

* Esto indica que el usuario "juan" pertenece a los siguientes grupos:
  1. **Grupo** juan: Este es el grupo principal del usuario.
  2. **Grupo** sudo: Permite al usuario "juan" ejecutar comandos como superusuario.
  3. **Grupo** docker: Permite al usuario "juan" usar Docker sin necesidad de ser superusuario.

4.68

a) Crear un grupo llamado "Bentas":

bash

sudo groupadd Bentas

b) Encontrar el GID del grupo:

bash

getent group Bentas

c) Cambiar el nombre del grupo "Bentas" a "Ventas":

bash

sudo groupmod -n Ventas Bentas

d) Cambiar el GID del grupo "Ventas" a 1500:

bash

sudo groupmod -g 1500 Ventas

e) Mostrar la línea correspondiente al grupo "Ventas" en el archivo /etc/group:

bash

grep Ventas /etc/group

f) Agregar al usuario creado durante la instalación al grupo "Ventas":

bash

sudo usermod -aG Ventas tu\_usuario

g) Mostrar la línea correspondiente al grupo "Ventas" en el archivo /etc/group:

bash

grep Ventas /etc/group

h) Eliminar el grupo "Ventas":

bash

sudo groupdel Ventas

i) La razón por la que se puede eliminar un grupo aunque tenga usuarios es que la eliminación de un grupo no elimina a los usuarios; solo elimina la asociación del grupo.

4.69

a) Localizar la línea correspondiente al usuario creado durante la instalación en /etc/shadow:

bash

grep tu\_usuario /etc/shadow

b) Localizar la línea correspondiente al grupo del usuario creado durante la instalación en /etc/gshadow:

bash

grep tu\_grupo /etc/gshadow

c) Localizar los usuarios bloqueados en /etc/shadow:

bash

grep '!'

4.70

a) Número máximo de días que una contraseña es válida:

bash

grep PASS\_MAX\_DAYS /etc/login.defs

b) Mínimo número de días permitido entre cambios de contraseña:

bash

grep PASS\_MIN\_DAYS /etc/login.defs

c) Número de días que se avisará con antelación antes de que una contraseña expire:

bash

grep PASS\_WARN\_AGE /etc/login.defs

d) Valor mínimo de UID para los usuarios creados con useradd:

bash

grep UID\_MIN /etc/login.defs

e) Valor máximo de UID para los usuarios creados con useradd:

bash

grep UID\_MAX /etc/login.defs

4.71

a) Crear un usuario de nombre "pepe" y asignarle una contraseña:

bash

sudo useradd pepe  
sudo passwd pepe

Utilizar el comando id para conocer su UID:

bash

id pepe

b) Utilizar el comando groups para conocer los grupos a los que pertenece "pepe":

bash

groups pepe

c) Crear el grupo "compras" y asignarle una contraseña:

bash

sudo groupadd compras  
sudo gpasswd compras

d) Localizar la línea correspondiente al grupo "compras" en el archivo /etc/gshadow:

bash

grep compras /etc/gshadow

e) Agregar al usuario "pepe" al grupo "compras":

bash

sudo usermod -aG compras pepe

f) Listar la línea correspondiente al grupo "compras" en el archivo /etc/group:

bash

grep compras /etc/group

g) Eliminar al usuario "pepe" del grupo "compras":

bash

sudo gpasswd -d pepe compras

h) Bloquear al usuario "pepe" e intentar iniciar sesión:

bash

sudo usermod -L pepe

Para comprobar que no puedes iniciar sesión, intenta iniciar sesión normalmente y verifica que se te niega el acceso. i) Localizar en el fichero /etc/shadow el símbolo "!" delante de la contraseña de "pepe":

bash

grep pepe /etc/shadow

j) Desbloquear al usuario "pepe":

bash

sudo usermod -U pepe

4.72

a) Listar la información de la cuenta de "pepe":

bash

chage -l pepe

b) Cambiar el día de expiración de la cuenta al 30 de Junio del 2025:

bash

sudo chage -E 2025-06-30 pepe

c) Cambiar el número mínimo de días entre cambios de clave a 15:

bash

sudo chage -m 15 pepe

d) Cambiar el número máximo de días para cambiar la clave a 30 días:

bash

sudo chage -M 30 pepe

e) Establecer el número de días de aviso a 1:

bash

sudo chage -W 1 pepe

f) Listar la información de la cuenta para ver los valores que has puesto:

bash

chage -l pepe

g) Crear un fichero de nombre "fich" en tu carpeta personal y convertir al usuario "pepe" en el propietario:

bash

touch ~/fich  
sudo chown pepe ~/fich

h) Cambiar el grupo propietario de "fich" por el grupo "compras":

bash

sudo chown :compras ~/fich

4.73

#### **a) ps –eo pid,rss, cmd**

sh

ps -eo pid,rss,cmd

Esto muestra los ID de proceso (PID), la cantidad de memoria residente (RSS) en KB y el comando asociado a cada proceso. La cantidad de memoria en KB asignada a esta orden se puede verificar sumando los valores de RSS de la salida.

#### **b) ps u**

sh

ps u

Esto muestra información detallada sobre los procesos, incluyendo el porcentaje de memoria usado por cada proceso. El porcentaje de memoria utilizado por el shell y la ejecución de la orden ps u se puede ver en la columna %MEM correspondiente a cada proceso.

#### **c) ps -f**

sh

ps -f

Muestra una lista completa con información detallada de los procesos. El identificador del proceso padre (PPID) de la ejecución del comando se encuentra en la columna PPID. Este ID generalmente corresponde al shell desde el cual se ejecutó el comando.

#### **d) ps -l**

sh

ps -l

Muestra una lista con información detallada incluyendo el estado de los procesos. Los procesos en ejecución tienen el estado R (running), mientras que los procesos detenidos tienen el estado T (stopped).

#### **e) Visualizar procesos de un usuario específico**

sh

ps -u nombre\_usuario

Reemplaza nombre\_usuario con el nombre del usuario que creaste durante la instalación.

#### **f) Visualizar procesos que no pertenecen a un usuario específico**

sh

ps -U nombre\_usuario

Reemplaza nombre\_usuario con el nombre del usuario que creaste durante la instalación. Esto muestra todos los procesos que no son de este usuario.

#### **g) Visualizar procesos ordenados por tiempo de ejecución**

sh

ps -eo pid,etime,cmd --sort=etime

Esto muestra los procesos ordenados por el tiempo de ejecución.

#### **h) ps r**

sh

ps

4.74

#### **a) top –u root**

top -u root

Esto visualiza los procesos del usuario root.

#### **b) Visualizar procesos del usuario autenticado con top**

sh

top -u $(whoami)

Muestra los procesos del usuario con el que estás autenticado.

#### **c) Ejecutar find y detener proceso con CTRL+Z**

sh

find / -name mifichero

Presiona CTRL+Z para detener el proceso.

#### **d) Utilizar kill para reanudar proceso**

sh

kill -CONT %n

Reemplaza %n con el número del trabajo que obtuviste con jobs.

#### **e) Utilizar kill para matar proceso**

sh

kill %n

4.75

#### **a) Ejecutar sleep 150 & y verificar jobs**

sh

sleep 150 &  
jobs

Esto muestra los trabajos en segundo plano y sus estados.

#### **b) fg %n**

sh

fg %n

Reemplaza %n con el número del trabajo. Esto trae el trabajo al primer plano.

#### **c) Utilizar CTRL+Z**

sh

sleep 150

Presiona CTRL+Z. Esto detiene el proceso y lo envía al fondo.

#### **d) Verificar jobs**

sh

jobs

Esto muestra el estado de los trabajos actuales.

#### **e) Ejecutar sleep 50 & y verificar jobs**

sh

sleep 50 &  
jobs

Esto muestra los trabajos en segundo plano y sus estados.

#### **f) Traer proceso al primer plano**

sh

fg %n

4.76

#### **a) Editar el archivo crontab para programar la ejecución de una orden**

sh

crontab -e

Añade la siguiente línea para ejecutar la orden "tar –uf copia.tar ventas" a las 13:50 todos los días:

sh

50 13 \* \* \* tar –uf copia.tar ventas

Guarda y cierra el archivo.

#### **b) Listar las tareas programadas de tu usuario**

sh

crontab -l

#### **c) Crear el fichero /etc/cron.deny y agregar tu usuario**

sh

sudo touch /etc/cron.deny  
echo "tu\_usuario" | sudo tee -a /etc/cron.deny  
crontab -e

Al intentar editar el crontab, obtendrás un mensaje de error diciendo que no tienes permiso porque estás en la lista de cron.deny.

#### **d) Crear el fichero /etc/cron.allow y agregar tu usuario**

sh

sudo touch /etc/cron.allow  
echo "tu\_usuario" | sudo tee -a /etc/cron.allow  
crontab -e

Ahora deberías poder editar el crontab, ya que tu usuario está permitido en cron.allow.

#### **e) Intentar editar el fichero crontab de otro usuario**

sh

sudo crontab -e -u otro\_usuario

Si no tienes permisos sudo, obtendrás un mensaje de error indicando que no tienes permiso para editar el crontab de otro usuario.

#### **f) Eliminar los ficheros cron.allow y cron.deny e intentar acceder al fichero crontab de otro usuario**

sh

sudo rm /etc/cron.allow /etc/cron.deny  
sudo crontab -e -u otro\_usuario

Si no existe ningún archivo cron.allow o cron.deny, deberías poder acceder al crontab del otro usuario si tienes permisos sudo.

#### **g) Acceder al directorio /var/spool/cron/crontabs**

sh

cd /var/spool/cron/crontabs  
ls

Aquí deberías ver el fichero de tareas programadas de tu usuario.

#### **h) Eliminar la tarea activa de tu usuario y verificar**

sh

crontab -e

Elimina la línea que corresponde a la tarea activa y guarda el archivo. Luego verifica:

sh

ls /var/spool/cron/crontabs

Deberías seguir viendo el fichero de tu usuario.

#### **i) Verificar la última ejecución de los scripts en cron.daily**

sh

sudo cat /var/spool/anacron/cron.daily

4.77

#### **a) Instalar at**

sh

sudo apt install at

#### **b) Programar la apertura de Firefox y gedit**

sh

echo "firefox & gedit" | at now

#### **c) Programar la lista de ficheros ocultos en una hora**

sh

echo "ls -la" | at now + 1 hour

#### **d) Programar la compresión del fichero "fich" a las 11:00 dentro de 5 días**

sh

echo "tar -czf fich.tar fich" | at 11:00 AM + 5 days

#### **e) Programar el apagado del equipo el 30 de junio a las 14:20**

sh

echo "shutdown -h now" | at 14:20 30 Jun

#### **f) Listar las tareas programadas con atq**

sh

atq

#### **g) Eliminar la tarea programada para comprimir el fichero "fich"**

sh

atrm número\_tarea

Reemplaza número\_tarea con el número de la tarea obtenida con atq.

#### **h) Verificar las tareas pendientes con at -l**

sh

at -l

#### **i) Añadir un usuario a /etc/at.allow y verificar**

sh

sudo nano /etc/at.allow

Añade un usuario que no sea el tuyo. Luego intenta:

sh

at now

4.78

#### **a) Lista las particiones del disco en el que está instalado el sistema operativo**

sh

lsblk

Esto listará todos los bloques de dispositivos y sus particiones.

#### **b) Crea en el segundo disco una partición de 5 GB**

Usa fdisk o parted para gestionar las particiones. Aquí hay un ejemplo usando fdisk:

sh

sudo fdisk /dev/sdb

Dentro de fdisk, sigue estos pasos:

1. Pulsa n para crear una nueva partición.
2. Selecciona p para una partición primaria.
3. Selecciona el número de la partición, por ejemplo 1.
4. Pulsa Enter para aceptar el valor predeterminado del primer sector.
5. Escribe +5G para establecer el tamaño de la partición a 5 GB.
6. Pulsa w para guardar los cambios y salir.

#### **c) Crea en el segundo disco una partición de 3 GB**

Repite los mismos pasos anteriores pero esta vez con +3G para la segunda partición:

sh

sudo fdisk /dev/sdb

1. Pulsa n para crear una nueva partición.
2. Selecciona p para una partición primaria.
3. Selecciona el número de la partición, por ejemplo 2.
4. Pulsa Enter para aceptar el valor predeterminado del primer sector.
5. Escribe +3G para establecer el tamaño de la partición a 3 GB.
6. Pulsa w para guardar los cambios y salir.

#### **d) Crea en el segundo disco una partición de 2 GB**

Repite los mismos pasos anteriores pero esta vez con +2G para la tercera partición:

sh

sudo fdisk /dev/sdb

1. Pulsa n para crear una nueva partición.
2. Selecciona p para una partición primaria.
3. Selecciona el número de la partición, por ejemplo 3.
4. Pulsa Enter para aceptar el valor predeterminado del primer sector.
5. Escribe +2G para establecer el tamaño de la partición a 2 GB.
6. Pulsa w para guardar los cambios y salir.

#### **e) Lista las particiones creadas en el segundo disco**

sh

lsblk

Esto mostrará las particiones creadas en el segundo disco (/dev/sdb).

#### **f) Guarda los cambios en el disco**

Esto ya se ha hecho al escribir los cambios en fdisk utilizando el comando w.

#### **g) Formatea la partición de 5 GB con el sistema de archivos ext4**

sh

sudo mkfs.ext4 /dev/sdb1

#### **h) Formatea la partición de 3 GB con el sistema de archivos ext3**

sh

sudo mkfs.ext3 /dev/sdb2

#### **i) Formatea la partición de 2 GB con el sistema de archivos ntfs**

sh

sudo mkfs.ntfs /dev/sdb3

#### **j) Desmonta la partición de 3 GB**

sh

sudo umount /dev/sdb2

#### **k) Formatea la partición de 3 GB con el sistema de archivos ext4**

sh

sudo mkfs.ext4 /dev/sdb2

#### **l) Comprueba que los sistemas de archivos se han creado correctamente**

sh

lsblk -f