

1.

Crear Particiones

Dispositivo	Inicio	Comienzo	Final	Sectores	Tamaño	Id	Tipo
/dev/sdc1		2048	31459327	31457280	15G	83	Linux
/dev/sdc2		31459328	73402367	41943040	20G	27	NTFS de WinRE oculta
/dev/sdc3		73402368	83886079	10483712	5G	82	Linux swap / Solaris

Formatear Particiones

```
root@alumno-VirtualBox:/home/alumno# mkfs /dev/sdc1
mke2fs 1.46.5 (30-Dec-2021)
Se está creando un sistema de ficheros con 3932160 bloques de 4k y 983040 nodos-i
UUID del sistema de ficheros: 1f1339a7-bed8-4cee-9cba-d503b667b11f
Respalos del superbloque guardados en los bloques:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208
Reservando las tablas de grupo: hecho
Escribiendo las tablas de nodos-i: hecho
Escribiendo superbloques y la información contable del sistema de archivos: 0/hecho
root@alumno-VirtualBox:/home/alumno# mkfs.
mkfs.bfs    mkfs.cramfs  mkfs.ext2    mkfs.ext3    mkfs.ext4    mkfs.fat     mkfs.minix  mkfs.msdos   mkfs.ntfs    mkfs.vfat
root@alumno-VirtualBox:/home/alumno# mkfs.ntfs /dev/sdc
sdc  sdc1  sdc2  sdc3
root@alumno-VirtualBox:/home/alumno# mkfs.ntfs /dev/sdc2
Cluster size has been automatically set to 4096 bytes.
Initializing device with zeroes: 10%
```

Montar particiones

```
root@alumno-VirtualBox:/home/alumno# mount /dev/sdc1 /home/alumno/Descargas/
root@alumno-VirtualBox:/home/alumno# mkdir /mnt/Windows
root@alumno-VirtualBox:/home/alumno# mount /dev/sdc2 /mnt/Windows/
root@alumno-VirtualBox:/home/alumno#
```

2.

Crear cuentas de usuario

```
usuario1:x:1001:1000::/home/usuario1/usuario1:/bin/bash
admin1:x:1002:1000::/home/admin1/admin1:/bin/bash
```

Crear la contraseña

```
root@alumno-VirtualBox:/home/alumno# passwd admin1
Nueva contraseña:
CONTRASEÑA INCORRECTA: La contraseña tiene menos de 8 caracteres
Vuelva a escribir la nueva contraseña:
Las contraseñas no coinciden.
Nueva contraseña:
CONTRASEÑA INCORRECTA: La contraseña no supera la verificación de diccionario - Está basada en una palabra del diccionario.
Vuelva a escribir la nueva contraseña:
Las contraseñas no coinciden.
Nueva contraseña:
CONTRASEÑA INCORRECTA: La contraseña no supera la verificación de diccionario - Es demasiado simple/sistemática.
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
root@alumno-VirtualBox:/home/alumno# passwd usuario1
Nueva contraseña:
CONTRASEÑA INCORRECTA: La contraseña no supera la verificación de diccionario - Es demasiado simple/sistemática.
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
root@alumno-VirtualBox:/home/alumno#
```

3.

Crear los grupos y añadir los usuarios

```
root@alumno-VirtualBox:/home/alumno# usermod -aG contabilidad usuario1
root@alumno-VirtualBox:/home/alumno# usermod -aG video usuario1
```

```
root@alumno-VirtualBox:/home/alumno# usermod -aG almacen admin1
root@alumno-VirtualBox:/home/alumno# usermod -aG administracion admin1
root@alumno-VirtualBox:/home/alumno# usermod -aG games admin1
```

4.

Crear los dos ficheros

```
root@alumno-VirtualBox:/home/alumno# echo "Contenido de la busqueda del directorio /dev" >prueba1
root@alumno-VirtualBox:/home/alumno# find /dev -type c -group tty 2>> errores >> prueba1

root@alumno-VirtualBox:/home/alumno# echo "Contenido de la busqueda del directorio /grub" > prueba2
root@alumno-VirtualBox:/home/alumno# find /grub -type d -iname "g*" 2>> errores >> prueba2
root@alumno-VirtualBox:/home/alumno#
```

5.

Modificar permisos de los ficheros

```
root@alumno-VirtualBox:/home# chmod u-w,g-rw,o-rw prueba1
```

```
root@alumno-VirtualBox:/home# chmod u-w,g-w,o-rw prueba2
```

6.

Modificar

```
root@alumno-VirtualBox:/home/alumno# chown usuario1:alumno prueba1
root@alumno-VirtualBox:/home/alumno# chown admin1:root prueba2
```

7.

Copiar los archivos

```
root@alumno-VirtualBox:/home/alumno# cp prueba1 /home/admin1/Proyectos
```

```
root@alumno-VirtualBox:/home/alumno# cp prueba2 /home/admin1/Proyectos
```

```
root@alumno-VirtualBox:/home/alumno# cp errores /home/admin1/Proyectos
```

Empaquetarlo

```
root@alumno-VirtualBox:/home/alumno# tar -czvf Proyectos.tgz Proyectos
```

copiarlo

```
root@alumno-VirtualBox:/home/alumno# cp Proyectos.tgz /home/usuario1/Proyectos_admin.tgz
```

8.

Sacar listado de procesos

```
root@alumno-VirtualBox:/home/alumno# ps -eo pid,pri,comm
```

PID	PRI	COMMAND
1	19	systemd
2	19	kthreadd
3	39	rcu_gp
4	39	rcu_par_gp
5	39	slub_flushwq
6	39	netns
8	39	kworker/0:0H-events_highpri
10	39	mm_percpu_wq
11	19	rcu_tasks_kthread
12	19	rcu_tasks_rude_kthread
13	19	rcu_tasks_trace_kthread
14	19	ksoftirqd/0
15	19	rcu_preempt
16	139	migration/0
17	90	idle_inject/0
19	19	cpuhp/0
20	19	kdevtmpfs
21	39	inet_frag_wq
22	19	kauditd
23	19	khungtaskd
24	19	kworker/u2:1-events_unbound
25	19	oom_reaper

```
root@alumno-VirtualBox:/home/alumno# ps -q 1000 -eo pid,comm
```

PID	COMMAND
5	slub_flushwq

9.

Estado de los servicios

```
root@alumno-VirtualBox:/home/alumno# systemctl status smb
Unit smb.service could not be found.
root@alumno-VirtualBox:/home/alumno# systemctl status cups
● cups.service - CUPS Scheduler
   Loaded: loaded (/lib/systemd/system/cups.service; enabled; preset: enabled)
   Active: active (running) since Fri 2024-02-02 08:21:12 CET; 1h 4min ago
 TriggeredBy: ● cups.socket
               ● cups.path
   Docs: man:cupsd(8)
  Main PID: 746 (cupsd)
   Status: "Scheduler is running..."
    Tasks: 4 (limit: 8225)
   Memory: 8.8M
      CPU: 101ms
   CGroup: /system.slice/cups.service
           └─746 /usr/sbin/cupsd -l
             └─772 /usr/lib/cups/notifier/dbus dbus://
               └─773 /usr/lib/cups/notifier/dbus dbus://
                 └─775 /usr/lib/cups/notifier/dbus dbus://

feb 02 08:21:11 alumno-VirtualBox systemd[1]: Starting CUPS Scheduler...
feb 02 08:21:12 alumno-VirtualBox systemd[1]: Started CUPS Scheduler.
root@alumno-VirtualBox:/home/alumno# systemctl status ssh
Unit ssh.service could not be found.
root@alumno-VirtualBox:/home/alumno#
```

Parar el servicio

```
root@alumno-VirtualBox:/home/alumno# systemctl stop cups
root@alumno-VirtualBox:/home/alumno#
```

10.

Ejecutar en segundo plano

```
root@alumno-VirtualBox:/home/alumno# ping -c 15 127.0.0.1 > /home/alumno/google_local &
[1] 5364
root@alumno-VirtualBox:/home/alumno#
```

```
root@alumno-VirtualBox:/home/alumno# ping -c www.google.es > /home/alumno/google_local &
[2] 5378
[1] Hecho ping -c 15 127.0.0.1 > /home/alumno/google_local
```

Comprobar el estado

```
[1]- Hecho ping -c 15 127.0.0.1 > /home/alumno/google_local
[2]+ Salida 1 ping -c 10 www.google.es > /home/alumno/google_local
root@alumno-VirtualBox:/home/alumno#
```

Primer plano

```
root@alumno-VirtualBox:/home/alumno# fg 5378
```

Finalizar el primer trabajo

```
root@alumno-VirtualBox:/home/alumno# kill - 9 5364
```

Actualizar los paquetes del sistema

```
root@alumno-VirtualBox:/home/alumno# at 22:00
warning: commands will be executed using /bin/sh
at Fri Feb  2 22:00:00 2024
at> apt-get update && apt-get -y upgrade
at> <EOT>
job 1 at Fri Feb  2 22:00:00 2024
root@alumno-VirtualBox:/home/alumno#
```

Buscar ficheros por tipo de enlace

```
root@alumno-VirtualBox:/home/alumno# at 22:00
warning: commands will be executed using /bin/sh
at Fri Feb  2 22:00:00 2024
at> find / -type l > /home/alumno/documentos/enlaces.txt 2 >> /home/alumno/documentos/errores_enl
aces.txt
at> <EOT>
job 2 at Fri Feb  2 22:00:00 2024
root@alumno-VirtualBox:/home/alumno#
```

12.

Scripts

Script1

```
#!/bin/bash
if [ $# -eq 0 ]; then
    echo "Uso: $0 directorio"
    exit 1
fi
directorio=$1
if [ $# -eq 0 ]; then
    echo "Uso: $1 directorio"
    exit 1
fi
directorio2=$2
function validar_directorio {
    if [ ! -d "$1" ]; then
        echo "$1 no es un directorio válido."
        return 1
    fi
    return 0
}
function validar_directorio {
    if [ ! -d "$2" ]; then
        echo "$2 no es un directorio válido."
        return 1
    fi
    return 0
}
while ! validar_directorio "$directorio"; do
    echo -n "Introduce un directorio válido: "
    read directorio
done
while ! validar_directorio "$directorio"; do
    echo -n "Introduce un directorio válido: "
    read directorio2
done
archivo_salida="misEjecutables"
> "$archivo_salida"
find "$directorio" -type f -executable -print >> "$archivo_salida"
find "$directorio2" -type f -executable -print >> "$archivo_salida"
echo "Contenido de $archivo_salida:"
cat "$archivo_salida"
numero_archivos_localizados=$(wc -l < "$archivo_salida")
echo "Se han localizado $numero_archivos_localizados archivos"
```

Script2

```
#!/bin/bash

declare -A usuarios
usuarios=( ["usuario1"]="contrasena123" ["usuario2"]="password456" )

while true; do
    read -p "Introduce tu nombre de usuario: " usuario

    if [[ ${usuarios[$usuario]}+_ ]]; then
        while true; do
            read -s -p "Introduce tu contraseña: " contrasena
            echo

            if [[ ${usuarios[$usuario]} == $contrasena ]]; then
                echo "Inicio de sesión exitoso."
                exit 0
            else
                echo "El password no coincide, inténtalo de nuevo."
            fi
        done
    else
        echo "Usuario no encontrado. Inténtalo de nuevo."
    fi
done
```