

LEO RUIZ GARCÍA PRACTICA DE IMPLANTACIÓN DE SISTEMAS OPERATIVOS

Instalar y configurar al menos 6 sistemas operativos. Tres de ellos serán FreeBSD, Openbsd y Minix. Además de ellos, se instalarán y configurarán tres sistemas operativos Linux (Debian, Suse, Centos, Kali Linux, Ubuntu, Red Hat, Fedora etc...) Debe entregarse.

1.- Modo de instalación del sistema operativo, cuanta ram necesita, que tamaño tiene el disco virtual, cuantas particiones tendrá. Que sistema de ficheros usa cada una de esas particiones.

Deben enumerarse todos los ítems que el sistema ha ido pidiendo cuando se ha ido instalando

2.- Mostrar el árbol de directorios de todos los sistemas operativos y anotar las coincidencias entre ellos.

3.- Indicar el nivel de ejecución en el que cada sistema operativo entra por defecto. (Ejecutar en una terminal la orden runlevel)

4.- Indicar si el sistema operativo dispone de servidor gráfico. Indicar si dispone de escritorio y cual es su nombre.

5.- Agregar un usuario al sistema.

6.- Ver si tiene conectividad con el exterior (orden ping)

7.- Mostrar la versión del núcleo en ejecución. Indicar dónde se encuentra dicho núcleo en el sistema de ficheros de cada s.o.

8.- Agregar un disco duro de 10 gigas a dicho sistema. Crear tres particiones en él y montar las particiones de dicho disco en /mnt/discos. En la carpeta discos se creará una carpeta por cada partición. Antes de montar nada se habrá procedido a crear los

sistemas de ficheros en cada partición. Debe documentarse sobre los sistemas de ficheros disponibles para cada S.O. De no existir /mnt/discos se crear

Para empezar, lo primero que haré sera instalar virtualbox para emular un ordenador dentro de un ordenador, aunque también se puede hacer en el propio ordenador flasheando un pendrive e instalandolo.

Una vez descargado virtualbox, lo siguiente que vamos hacer es descargar las isos que necesitemos, kali, freebsd, minx etc... Después de descargarlo vamos a configurar la máquina virtual. Lo primero que hay que hacer para crear una máquina virtual será crear un vdi que es un disco duro virtual en el cual se guardará el sistema operativo. Y una vez creado, lo que hay que hacer es agregar una iso de las que ya están descargadas. Estos pasos habrá que hacerlo para cada sistema que instalemos en el virtualbox

Kali linux

Para instalar este sistema hay que hacer los pasos anteriores. Una vez iniciado eset sistema, seguiremos los pasos que nos dice el sistema de instalación.

1.- Modo de instalación del sistema operativo, cuanta ram necesita, que tamaño tiene el disco virtual, cuantas particiones tendrá. Que sistema de ficheros usa cada una de esas particiones.

Deben enumerarse todos los ítems que el sistema ha ido pidiendo cuando se ha ido instalando

Para este sistema pondremos 16 gb de almacenamiento interno reservado dinámicamente, luego le ponemos 2 gb de ram, tendrá solo 1 partición más las que se instalan predeterminadas. Para comprobar la particiones que tiene el sistema usaré el comando fdisk -l y para saber que sistema de archivo tiene usare el comando file -s y la partición que queremos ver.



A la hora de instalarlo, cuando me preguntan varias opciones, elijo la que viene predeterminada, y creó usuario si me lo pide

```
leo@kali:~$ sudo fdisk -l
Disk /dev/sda: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x7ada1b92

Device      Boot    Start        End    Sectors    Size Id Type
/dev/sda1   *         2048    18970623   18968576     9G 83 Linux
/dev/sda2             18972670   20969471   1996802     975M  5 Extended
/dev/sda5             18972672   20969471   1996800     975M 82 Linux swap / Solaris

leo@kali:~$ file -s /dev/sda1
/dev/sda1: no read permission
leo@kali:~$ sudo file -s /dev/sda1
/dev/sda1: Linux rev 1.0 ext4 filesystem data, UUID=177a5083-c49c-4d4e-ae53-a61d81656ff1 (needs journal recovery) (extents) (64bit) (large files) (huge files)
```

Una vez instalado y haber configurado los parámetros que nos indica el sistemas de instalación, vamos a hacer la comprobaciones necesarias

2.- Mostrar el árbol de directorios de todos los sistemas operativos y anotar las coincidencias entre ellos.

Para mostrar el árbol de directorio usamos este comando:

ls -l /

```
leo@kali:~$ ls -l /
total 68
lrwxrwxrwx 1 root root 7 Sep 25 03:28 bin -> usr/bin
drwxr-xr-x 3 root root 4096 Sep 25 04:19 boot
drwxr-xr-x 17 root root 3240 Sep 25 05:41 dev
drwxr-xr-x 157 root root 12288 Sep 25 05:41 etc
drwxr-xr-x 3 root root 4096 Sep 25 04:16 home
lrwxrwxrwx 1 root root 33 Sep 25 03:29 initrd.img -> boot/initrd.img-5
.7.0-kali1-amd64
lrwxrwxrwx 1 root root 33 Sep 25 03:29 initrd.img.old -> boot/initrd.i
mg-5.7.0-kali1-amd64
lrwxrwxrwx 1 root root 7 Sep 25 03:28 lib -> usr/lib
lrwxrwxrwx 1 root root 9 Sep 25 03:28 lib32 -> usr/lib32
lrwxrwxrwx 1 root root 9 Sep 25 03:28 lib64 -> usr/lib64
lrwxrwxrwx 1 root root 10 Sep 25 03:28 libx32 -> usr/libx32
drwx----- 2 root root 16384 Sep 25 03:28 lost+found
drwxr-xr-x 4 root root 4096 Sep 25 03:28 media
drwxr-xr-x 2 root root 4096 Sep 25 03:28 mnt
drwxr-xr-x 2 root root 4096 Sep 25 03:28 opt
dr-xr-xr-x 184 root root 0 Sep 25 05:41 proc
drwx----- 3 root root 4096 Sep 25 05:41 root
drwxr-xr-x 33 root root 800 Sep 25 05:42 run
lrwxrwxrwx 1 root root 8 Sep 25 03:28 sbin -> usr/sbin
drwxr-xr-x 3 root root 4096 Sep 25 04:03 srv
dr-xr-xr-x 13 root root 0 Sep 25 05:41 sys
drwxrwxrwt 13 root root 4096 Sep 25 05:42 tmp
drwxr-xr-x 14 root root 4096 Sep 25 03:38 usr
drwxr-xr-x 12 root root 4096 Sep 25 03:37 var
lrwxrwxrwx 1 root root 30 Sep 25 03:29 vmlinuz -> boot/vmlinuz-5.7.0-k
ali1-amd64
lrwxrwxrwx 1 root root 30 Sep 25 03:29 vmlinuz.old -> boot/vmlinuz-5.7
.0-kali1-amd64
leo@kali:~$ uname -a
Linux kali 5.7.0-kali1-amd64 #1 SMP Debian 5.7.6-1kali2 (2020-07-01) x86_64 GNU/Linux
leo@kali:~$
```

3.- Indicar el nivel de ejecución en el que cada sistema operativo entra por defecto. (Ejecutar en una terminal la orden runlevel)

Para ver el nivel de ejecución del sistema operativo usamos este comando:
runlevel

```
root@kali:/home/leo# runlevel
N 5
```

4.- Indicar si el sistema operativo dispone de servidor gráfico. Indicar si dispone de escritorio y cual es su nombre.

Este sistema es de entorno gráfico y dispone de escritorio. Para saber que tipo o que nombre de entorno gráfico se usa el comando:

ls /usr/bin/*session

```
root@kali:/home/leo# ls /usr/bin/*session
/usr/bin/dbus-run-session /usr/bin/xfce4-session
```

con lo cual el nombre del entorno gráfico que tiene es xfce4

Agregar usuario:

5.- Agregar un usuario al sistema.

Para agregar un usuario usamos el comando:

adduser + nombre de usuario o useradd + nombre de usuario y configuramos su contraseña

```
root@kali:/home/leo# adduser practica
Adding user `practica' ...
Adding new group `practica' (1001) ...
Adding new user `practica' (1001) with group `practica' ...
Creating home directory `/home/practica' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for practica
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
root@kali:/home/leo#
```

Para comprobar que el usuario está creado, hay que entrar al fichero /etc/passwd. Y para ver ese fichero sin tener que entrar en él usamos el comando cat /etc/passwd

```
geoclue:x:130:138::/var/lib/geoclue:/usr/sbin/nologin
lightdm:x:131:139:Light Display Manager:/var/lib/lightdm:/bin/false
king-phisher:x:132:140::/var/lib/king-phisher:/usr/sbin/nologin
leo:x:1000:1000:Leo Ruiz Garcia,,,:/home/leo:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:./usr/sbin/nologin
practica:x:1001:1001,,,:/home/practica:/bin/bash
```

Con lo cual, vemos que el usuario práctica está creado y tiene su propio directorio.

6.- Ver si tiene conectividad con el exterior (orden ping)

para comprobar si hay conexión a internet, hay que hacer por ejemplo una conexión con algo de fuera como por ejemplo www.google.es , usaremos este comando:

ping www.google.es, y si este comando nos da ping, significa que hay conexión a internet

```
root@kali:/home/leo# ping www.google.es
PING www.google.es (216.58.201.131) 56(84) bytes of data.
64 bytes from mad06s25-in-f3.1e100.net (216.58.201.131): icmp_seq=1 ttl=117
  time=9.11 ms
64 bytes from mad06s25-in-f3.1e100.net (216.58.201.131): icmp_seq=2 ttl=117
  time=17.3 ms
64 bytes from mad06s25-in-f3.1e100.net (216.58.201.131): icmp_seq=3 ttl=117
  time=10.1 ms
^C
--- www.google.es ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 9.112/12.169/17.268/3.629 ms
```

Para mostrar la version de nucleo usamos el comando:

uname -a

```
root@kali:/home/leo# uname -a
Linux kali 5.7.0-kali1-amd64 #1 SMP Debian 5.7.6-1kali2 (2020-07-01) x86_64
GNU/Linux
```

Y para ver donde se encuentra el núcleo usamos el comando:

7.- Mostrar la versión del núcleo en ejecución. Indicar dónde se encuentra dicho núcleo en el sistema de ficheros de cada s.o.

ls -l /

```
root@kali:/home/leo# ls -l /
total 68
lrwxrwxrwx 1 root root 7 Sep 25 03:28 bin -> usr/bin
drwxr-xr-x 3 root root 4096 Sep 25 04:19 boot
drwxr-xr-x 17 root root 3240 Sep 29 05:13 dev
drwxr-xr-x 157 root root 12288 Sep 29 05:13 etc
drwxr-xr-x 4 root root 4096 Sep 25 06:05 home
lrwxrwxrwx 1 root root 33 Sep 25 03:29 initrd.img -> boot/initrd.img-5
.7.0-kali1-amd64
lrwxrwxrwx 1 root root 33 Sep 25 03:29 initrd.img.old -> boot/initrd.i
mg-5.7.0-kali1-amd64
lrwxrwxrwx 1 root root 7 Sep 25 03:28 lib -> usr/lib
lrwxrwxrwx 1 root root 9 Sep 25 03:28 lib32 -> usr/lib32
lrwxrwxrwx 1 root root 9 Sep 25 03:28 lib64 -> usr/lib64
lrwxrwxrwx 1 root root 10 Sep 25 03:28 libx32 -> usr/libx32
drwx----- 2 root root 16384 Sep 25 03:28 lost+found
drwxr-xr-x 4 root root 4096 Sep 25 03:28 media
drwxr-xr-x 2 root root 4096 Sep 25 03:28 mnt
drwxr-xr-x 2 root root 4096 Sep 25 03:28 opt
dr-xr-xr-x 187 root root 0 Sep 29 05:13 proc
drwx----- 3 root root 4096 Sep 29 05:13 root
drwxr-xr-x 33 root root 800 Sep 29 05:14 run
lrwxrwxrwx 1 root root 8 Sep 25 03:28 sbin -> usr/sbin
drwxr-xr-x 3 root root 4096 Sep 25 04:03 srv
dr-xr-xr-x 13 root root 0 Sep 29 05:13 sys
drwxrwxrwt 13 root root 4096 Sep 29 05:14 tmp
drwxr-xr-x 14 root root 4096 Sep 25 03:38 usr
drwxr-xr-x 12 root root 4096 Sep 25 03:37 var
lrwxrwxrwx 1 root root 30 Sep 25 03:29 vmlinuz -> boot/vmlinuz-5.7.0-k
ali1-amd64
lrwxrwxrwx 1 root root 30 Sep 25 03:29 vmlinuz.old -> boot/vmlinuz-5.7
0-kali1-amd64
```

8.-Agregar un disco duro de 10 gigas a dicho sistema. Crear tres particiones en él y montar las particiones de dicho disco en /mnt/discos. En la carpeta discos se creará una carpeta por cada partición. Antes de montar nada se habrá procedido a crear los sistemas de ficheros en cada partición. Debe documentarse sobre los sistemas de ficheros disponibles para cada S.O. De no existir /mnt/discos se creará.

Para hacer las particiones, usará el comando fdisk

```
leo@kali:~$ sudo fdisk /dev/sdb

Welcome to fdisk (util-linux 2.35.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xc6263c86.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-20971519, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-20971519, default 20971519): 300000

Created a new partition 1 of type 'Linux' and of size 145.5 MiB.
```

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sdb1		2048	300000	297953	145.5M	83	Linux
/dev/sdb2		301056	600000	298945	146M	83	Linux
/dev/sdb3		600064	20971519	20371456	9.7G	83	Linux

A los discos primeros les puse 300.000 y a el ultimo le puse lo que sobraba

A continuación formatear las particiones y poder xt4 con el comando mkfs.ext4 /dev/sdb para cada 1 de las tres particiones

```
root@kali:/home/leo# mkfs.ext4 /dev/sdb2
mke2fs 1.45.6 (20-Mar-2020)
Creating filesystem with 149472 1k blocks and 37392 inodes
Filesystem UUID: 359992ff-2307-4160-9781-6cfac8bf93db
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

root@kali:/home/leo# mkfs.ext4 /dev/sdb3
mke2fs 1.45.6 (20-Mar-2020)
Creating filesystem with 2546432 4k blocks and 637728 inodes
Filesystem UUID: 6e2840ad-98b7-482d-b83c-0349fac33f95
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```


A la hora de montar el disco usare el comando mount /la partición que quiero montar /donde la quiero montar

```
root@kali:/home/leo# mount /dev/sdb1 /mnt/discos/  
mount: /mnt/discos: wrong fs type, bad option, bad superblock on /dev/sd  
missing codepage or helper program, or other error.
```

Ubuntu 20

1.- Modo de instalación del sistema operativo, cuanta ram necesita, que tamaño tiene el disco virtual, cuantas particiones tendrá. Que sistema de ficheros usa cada una de esas particiones.

Deben enumerarse todos los ítems que el sistema ha ido pidiendo cuando se ha ido instalando

A la hora de instalarlo, cuando me preguntan varias opciones, elijo la que viene predeterminada, y creó usuario si me lo pide



Para este sistema pondremos 16 gb de almacenamiento interno reservado dinámicamente, luego le ponemos 2 gb de ram, tendrá solo 1 partición más las que se instalan predeterminadas. Para comprobar la particiones que tiene el sistema usare el comando `fdisk -l` y para saber que sistema de archivo tiene usare el comando `file -s` y la partición que queremos ver.

Particiones:

Dispositivo	Inicio	Comienzo	Final	Sectores	Tamaño	Id	Tipo
/dev/sda1	*	2048	1050623	1048576	512M	b W95	FAT32
/dev/sda2		1052670	20969471	19916802	9,5G	5	Extendida
/dev/sda5		1052672	20969471	19916800	9,5G	83	Linux

```
leo@leo-VirtualBox:~$ sudo file -s /dev/sda1
/dev/sda1: DOS/MBR boot sector, code offset 0x58+2, OEM-ID "mkfs.fat",
```

```
leo@leo-VirtualBox:~$ sudo file -s /dev/sda2
/dev/sda2: DOS/MBR boot sector; partition 1 :
```

```
leo@leo-VirtualBox:~$ sudo file -s /dev/sda5
/dev/sda5: Linux rev 1.0 ext4 filesystem data,
```

2.- Mostrar el árbol de directorios de todos los sistemas operativos y anotar las coincidencias entre ellos.

Para mostrar el árbol de directorio usamos este comando:

```
ls -l /
```

```
leo@leo-VirtualBox:~$ ls -l /
total 459344
lrwxrwxrwx   1 root root          7 sep 25 09:51 bin -> usr/bin
drwxr-xr-x   4 root root    4096 sep 25 10:23 boot
drwxrwxr-x   2 root root    4096 sep 25 10:00 cdrom
drwxr-xr-x  18 root root    4040 sep 29 12:06 dev
drwxr-xr-x 130 root root   12288 sep 25 11:38 etc
drwxr-xr-x   4 root root    4096 sep 25 11:36 home
lrwxrwxrwx   1 root root          7 sep 25 09:51 lib -> usr/lib
lrwxrwxrwx   1 root root          9 sep 25 09:51 lib32 -> usr/lib32
lrwxrwxrwx   1 root root          9 sep 25 09:51 lib64 -> usr/lib64
lrwxrwxrwx   1 root root         10 sep 25 09:51 libx32 -> usr/libx32
drwx-----  2 root root   16384 sep 25 09:51 lost+found
drwxr-xr-x   2 root root    4096 jul 31 18:27 media
drwxr-xr-x   2 root root    4096 jul 31 18:27 mnt
d-----    1 root root    4096 jul 31 18:27 opt
d-----    1 root root    4096 jul 31 18:27 proc
drwx-----  4 root root    4096 sep 25 11:38 root
drwxr-xr-x  30 root root     800 sep 29 12:06 run
lrwxrwxrwx   1 root root          8 sep 25 09:51 sbin -> usr/sbin
drwxr-xr-x   8 root root    4096 sep 25 11:16 snap
drwxr-xr-x   2 root root    4096 jul 31 18:27 srv
-rw-----  1 root root 470287360 sep 25 09:51 swapfile
dr-xr-xr-x  13 root root         0 sep 29 12:06 sys
drwxrwxrwt  18 root root    4096 sep 29 12:09 tmp
drwxr-xr-x  14 root root    4096 jul 31 18:28 usr
drwxr-xr-x  14 root root    4096 jul 31 18:35 var
```

3.- Indicar el nivel de ejecución en el que cada sistema operativo entra por defecto. (Ejecutar en una terminal la orden runlevel)

```
leo@leo-VirtualBox:~$ runlevel
N 5
```

4.- Indicar si el sistema operativo dispone de servidor gráfico. Indicar si dispone de escritorio y cual es su nombre.

Este sistema es de entorno gráfico y dispone de escritorio. Para saber que tipo o que nombre de entorno gráfico se usa el comando:

`ls /usr/bin/*session`

```
leo@leo-VirtualBox:~$ ls /usr/bin/*session
/usr/bin/dbus-run-session  /usr/bin/gnome-session-custom-session
/usr/bin/gnome-session
```

5.- Agregar un usuario al sistema.

Para agregar un usuario usamos el comando:

`adduser + nombre de usuario` o `useradd + nombre de usuario` y configuramos su contraseña

```
root@leo-VirtualBox:/home/leo# adduser practica
Añadiendo el usuario 'practica' ...
Añadiendo el nuevo grupo 'practica' (1001) ...
Añadiendo el nuevo usuario 'practica' (1001) con grupo 'practica' ...
El directorio personal '/home/practica' ya existe. No se copiará desde '/etc/skel'.
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
Cambiano la información de usuario para practica
Introduzca el nuevo valor, o presione INTRO para el predeterminado
Nombre completo []:
Número de habitación []:
Teléfono del trabajo []:
Teléfono de casa []:
Otro []:
¿Es correcta la información? [S/n] s
root@leo-VirtualBox:/home/leo#
```

Y para comprobar que el usuario se ha creado correctamente hay que usar este comando:

`cat /etc/passwd`

```
geoclue:x:122:127::/var/lib/geoclue:/usr/sbin/nologin
pulse:x:123:128:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534::/run/gnome-initial-setup:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
leo:x:1000:1000:leo,,,:/home/leo:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
practica:x:1001:1001:,,,:/home/practica:/bin/bash
```

Con lo cual, vemos que el usuario práctica está creado y tiene su propio directorio.

6.- Ver si tiene conectividad con el exterior (orden ping)

para comprobar si hay conexión a internet, hay que hacer por ejemplo una conexión con algo de fuera como por ejemplo `www.google.es`, usaremos este comando:

`ping www.google.es`, y si este comando nos da ping, significa que hay conexión a internet

```
root@leo-VirtualBox:/home/leo# ping www.google.es
PING www.google.es (216.58.201.131) 56(84) bytes of data.
64 bytes from mad06s25-in-f3.1e100.net (216.58.201.131): icmp_seq=1 ttl=117 time=8.74 ms
64 bytes from mad06s25-in-f3.1e100.net (216.58.201.131): icmp_seq=2 ttl=117 time=10.2 ms
64 bytes from mad06s25-in-f3.1e100.net (216.58.201.131): icmp_seq=3 ttl=117 time=53.6 ms
^C
--- www.google.es ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 8.737/24.186/53.612/20.815 ms
root@leo-VirtualBox:/home/leo#
```

```
root@leo-VirtualBox:/home/leo# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::fa0b:e1e7:e9f6:910 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:ef:60:21 txqueuelen 1000 (Ethernet)
    RX packets 26873 bytes 34108128 (34.1 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3380 bytes 236914 (236.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Bucle local)
    RX packets 211 bytes 18179 (18.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 211 bytes 18179 (18.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@leo-VirtualBox:/home/leo#
```

7.- Mostrar la versión del núcleo en ejecución. Indicar dónde se encuentra dicho núcleo en el sistema de ficheros de cada s.o.

```
root@leo-VirtualBox:/home/leo# uname -a
Linux leo-VirtualBox 5.4.0-48-generic #52-Ubuntu SMP Thu Sep 10 10:58:49 UTC 20
20 x86_64 x86_64 x86_64 GNU/Linux
```

```

root@leo-VirtualBox:/home/leo# ls /boot
config-5.4.0-42-generic      memtest86+.elf
config-5.4.0-48-generic      memtest86+_multiboot.bin
efi                          System.map-5.4.0-42-generic
grub                         System.map-5.4.0-48-generic
initrd.img                  vmlinuz
initrd.img-5.4.0-42-generic  vmlinuz-5.4.0-42-generic
initrd.img-5.4.0-48-generic  vmlinuz-5.4.0-48-generic
initrd.img.old              vmlinuz.old
memtest86+.bin

```

8.-Agregar un disco duro de 10 gigas a dicho sistema. Crear tres particiones en él y montar las particiones de dicho disco en /mnt/discos. En la carpeta discos se creará una carpeta por cada partición. Antes de montar nada se habrá procedido a crear los sistemas de ficheros en cada partición. Debe documentarse sobre los sistemas de ficheros disponibles para cada S.O. De no existir /mnt/discos se creará.

Este es el disco que vamos a dar particiones

```

Disco /dev/sda: 10 GiB, 10737418240 bytes, 20971520 sectores
Disk model: VBOX HARDDISK
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0x779d91a0

Dispositivo Inicio Comienzo      Final Sectores Tamaño Id Tipo
/dev/sda1 *          2048 1050623 1048576 512M b W95 FAT32
/dev/sda2             1052670 20969471 19916802 9,5G 5 Extendida
/dev/sda5             1052672 20969471 19916800 9,5G 83 Linux

Disco /dev/sdb: 10 GiB, 10737418240 bytes, 20971520 sectores
Disk model: VBOX HARDDISK
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
root@leo-VirtualBox:/home/leo#

```

Para la primera partición será de 300000 , la segunda de 600000 y la tercera de lo que sobra

```

Dispositivo Inicio Comienzo      Final Sectores Tamaño Id Tipo
/dev/sdb1             2048 300000 297953 145,5M 83 Linux
/dev/sdb2             301056 600000 298945 146M 83 Linux
/dev/sdb3             600064 20971519 20371456 9,7G 83 Linux

```


A continuación formatear las particiones y poder ext4 con el comando `mkfs.ext4 /dev/sdb` para cada una de las tres particiones

```
root@leo-VirtualBox:/home/leo# mkfs.ext4 /dev/sdb2
mke2fs 1.45.5 (07-Jan-2020)
Se está creando un sistema de ficheros con 37368 bloques de 4k y 37376 nodos-i
UUID del sistema de ficheros: af2fbc20-d381-480e-913d-d7f69e996f20
Respalos del superbloque guardados en los bloques:
    32768

Reservando las tablas de grupo: hecho
Escribiendo las tablas de nodos-i: hecho
Creando el fichero de transacciones (4096 bloques): hecho
Escribiendo superbloques y la información contable del sistema de archivos:hech
o

root@leo-VirtualBox:/home/leo# mkfs.ext4 /dev/sdb3
mke2fs 1.45.5 (07-Jan-2020)
Se está creando un sistema de ficheros con 2546432 bloques de 4k y 637728 nodos
-i
UUID del sistema de ficheros: b163b26f-1ced-4af6-9898-cbe7b378ed13
Respalos del superbloque guardados en los bloques:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Reservando las tablas de grupo: hecho
Escribiendo las tablas de nodos-i: hecho
Creando el fichero de transacciones (16384 bloques): hecho
Escribiendo superbloques y la información contable del sistema de archivos: 0/
hecho
```

A la hora de montar el disco usare el comando `mount` /la partición que quiero montar /donde la quiero montar

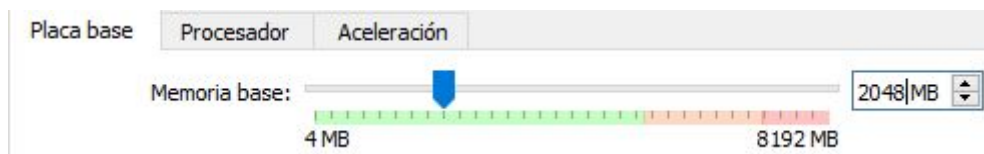
```
root@leo-VirtualBox:/home/leo# mount /dev/sdb1 /mnt/discos
mount: /mnt/discos: /dev/sdb1 ya está montado en /mnt/discos.
root@leo-VirtualBox:/home/leo# mount /dev/sdb2 /mnt/discos
root@leo-VirtualBox:/home/leo# mount /dev/sdb3 /mnt/discos
root@leo-VirtualBox:/home/leo#
```

fedora

1.- Modo de instalación del sistema operativo, cuanta ram necesita, que tamaño tiene el disco virtual, cuantas particiones tendrá. Que sistema de ficheros usa cada una de esas particiones.

Deben enumerarse todos los ítems que el sistema ha ido pidiendo cuando se ha ido instalando

En la instalación todo lo que me pregunto el menú de instalación lo deje predeterminado



Para este sistema pondremos 21 gb de almacenamiento interno reservado dinámicamente, luego le ponemos 2 gb de ram, tendrá solo 1 partición más las que se instalan predeterminadas. Para comprobar la particiones que tiene el sistema usará el comando `fdisk -l` y para saber que sistema de archivo tiene usare el comando `file -s` y la partición que queremos ver.

```
Disposit.  Inicio Comienzo  Final Sectores Tamaño Id Tipo
/dev/sda1  *          2048  2099199   2097152      1G 83 Linux
/dev/sda2          2099200 45492223 43393024   20,7G 8e Linux LVM
```

```
[root@localhost ~]# file -s /dev/sda1
/dev/sda1: Linux rev 1.0 ext4 filesystem data, UUID=ad4acd8e-65f9-4279-88fe-7c387771431b (needs j
ournal recovery) (extents) (64bit) (large files) (huge files)
[root@localhost ~]# file -s /dev/sda2
/dev/sda2: LVM2 PV (Linux Logical Volume Manager), UUID: lYnigi-pQb5-lU0c-Wsss-ElcK-00UW-W8f2oQ,
size: 22217228288
[root@localhost ~]#
```


2.- Mostrar el árbol de directorios de todos los sistemas operativos y anotar las coincidencias entre ellos.

Para mostrar el árbol de directorio usamos este comando:

```
ls -l /
```

```
[root@localhost ~]# ls -l /
total 64
lrwxrwxrwx. 1 root root 7 ene 28 2020 bin -> usr/bin
dr-xr-xr-x. 7 root root 4096 sep 25 11:03 boot
drwxr-xr-x. 20 root root 3860 oct 1 11:05 dev
drwxr-xr-x. 155 root root 12288 oct 1 11:05 etc
drwxr-xr-x. 5 root root 4096 sep 27 15:13 home
lrwxrwxrwx. 1 root root 7 ene 28 2020 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 ene 28 2020 lib64 -> usr/lib64
drwx----- 2 root root 16384 abr 23 00:28 lost+found
drwxr-xr-x. 2 root root 4096 ene 28 2020 media
drwxr-xr-x. 2 root root 4096 ene 28 2020 mnt
drwxr-xr-x. 2 root root 4096 ene 28 2020 opt
dr-xr-xr-x. 232 root root 0 oct 1 11:04 proc
dr-xr-xr-x. 16 root root 4096 oct 1 11:09 root
drwxr-xr-x. 46 root root 1300 oct 1 11:10 run
lrwxrwxrwx. 1 root root 8 ene 28 2020 sbin -> usr/sbin
drwxr-xr-x. 2 root root 4096 ene 28 2020 srv
dr-xr-xr-x. 13 root root 0 oct 1 11:04 sys
drwxrwxrwt. 16 root root 380 oct 1 11:10 tmp
drwxr-xr-x. 12 root root 4096 abr 23 00:35 usr
drwxr-xr-x. 20 root root 4096 abr 23 00:44 var
-rw----- 1 root root 0 oct 1 11:09 VBox.log
[root@localhost ~]#
```

3.- Indicar el nivel de ejecución en el que cada sistema operativo entra por defecto. (Ejecutar en una terminal la orden runlevel)

```
[root@localhost ~]# runlevel
N 5
```

4.-Indicar si el sistema operativo dispone de servidor gráfico. Indicar si dispone de escritorio y cual es su nombre.

Este sistema es de entorno gráfico y dispone de escritorio. Para saber que tipo o que nombre de entorno gráfico se usa el comando:

```
ls /usr/bin/*session
```

```
[root@localhost ~]# ls /usr/bin/*session
/usr/bin/dbus-run-session /usr/bin/plasma_session /usr/bin/tssstartauthsession
[root@localhost ~]#
```

Con lo cual esta información nos dice que este sistema es KDE plasma

5.- Agregar un usuario al sistema.

Para agregar un usuario usamos el comando:

adduser + nombre de usuario o useradd + nombre de usuario y configuramos su contraseña

```
[root@localhost ~]# useradd practicaLeo
```

```
practicaLeo:x:1005:1005::/home/practicaLeo:/bin/bash
```

Con lo cual, vemos que el usuario práctica está creado y tiene su propio directorio.

6.- Ver si tiene conectividad con el exterior (orden ping)

para comprobar si hay conexión a internet, hay que hacer por ejemplo una conexión con algo de fuera como por ejemplo www.google.es , usaremos este comando:

ping www.google.es, y si este comando nos da ping, significa que hay conexión a internet

```
[root@localhost ~]# ping www.google.es
PING www.google.es (216.58.201.131) 56(84) bytes of data.
64 bytes from mad06s25-in-f3.1e100.net (216.58.201.131): icmp_seq=1 ttl=117 time=9.82 ms
64 bytes from mad06s25-in-f3.1e100.net (216.58.201.131): icmp_seq=2 ttl=117 time=9.39 ms
64 bytes from mad06s25-in-f3.1e100.net (216.58.201.131): icmp_seq=3 ttl=117 time=9.58 ms
^C
--- www.google.es ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 9.394/9.596/9.816/0.172 ms
[root@localhost ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::af6f:e012:2902:4b0c prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:e1:2c:0a txqueuelen 1000 (Ethernet)
    RX packets 187281 bytes 224905304 (214.4 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 36572 bytes 2259226 (2.1 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 10 bytes 1564 (1.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 10 bytes 1564 (1.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@localhost ~]#
```

7.- Mostrar la versión del núcleo en ejecución. Indicar dónde se encuentra dicho núcleo en el sistema de ficheros de cada s.o.

```
[root@localhost ~]# uname -a
Linux localhost.localdomain 5.6.6-300.fc32.x86_64 #1 SMP Tue Apr 21 13:44:19 UTC 2020 x86_64 x86_64 GNU/Linux
[root@localhost ~]# ls /boot
config-5.6.6-300.fc32.x86_64
efi
elf-memtest86+-5.01
extlinux
grub2
initramfs-0-rescue-b77674af890f4a3e9a36365ba735a2f0.img
initramfs-5.6.6-300.fc32.x86_64.img
loader
lost+found
memtest86+-5.01
System.map-5.6.6-300.fc32.x86_64
vmlinuz-0-rescue-b77674af890f4a3e9a36365ba735a2f0
vmlinuz-5.6.6-300.fc32.x86_64
[root@localhost ~]#
```

8.-Agregar un disco duro de 10 gigas a dicho sistema. Crear tres particiones en él y montar las particiones de dicho disco en /mnt/discos. En la carpeta discos se creará una carpeta por cada partición. Antes de montar nada se habrá procedido a crear los sistemas de ficheros en cada partición. Debe documentarse sobre los sistemas de ficheros disponibles para cada S.O. De no existir /mnt/discos se creará.

Este es el disco que vamos a dar particiones

```
Disco /dev/sdb: 10 GiB, 10737418240 bytes, 20971520 sectores
Modelo de disco: VBOX HARDDISK
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
```

Disposit.	Inicio	Comienzo	Final	Sectores	Tamaño	Id	Tipo
/dev/sdb1		2048	300000	297953	145,5M	83	Linux
/dev/sdb2		301056	600000	298945	146M	83	Linux
/dev/sdb3		600064	20971519	20371456	9,7G	83	Linux

A los discos primeros les puse 300.000 y a el ultimo le puse lo que sobraba

A continuación formatear las particiones y poder xt4 con el comando mkfs.ext4 /dev/sdb para cada 1 de las tres particiones

```
[root@localhost ~]# mkfs.ext4 /dev/sdb2
mke2fs 1.45.5 (07-Jan-2020)
Se está creando un sistema de ficheros con 149472 bloques de 1k y 37392 nodos-i
UUID del sistema de ficheros: 7f2c70d9-dbe4-4e65-aebd-a92f382f6ed2
Respalos del superbloque guardados en los bloques:
    8193, 24577, 40961, 57345, 73729

Reservando las tablas de grupo: hecho
Escribiendo las tablas de nodos-i: hecho
Creando el fichero de transacciones (4096 bloques): hecho
Escribiendo superbloques y la información contable del sistema de ficheros: hecho

[root@localhost ~]# mkfs.ext4 /dev/sdb3
mke2fs 1.45.5 (07-Jan-2020)
Se está creando un sistema de ficheros con 2546432 bloques de 4k y 637728 nodos-i
UUID del sistema de ficheros: 47e69261-03fe-43a3-89de-d725a10fc456
Respalos del superbloque guardados en los bloques:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Reservando las tablas de grupo: hecho
Escribiendo las tablas de nodos-i: hecho
Creando el fichero de transacciones (16384 bloques): hecho
Escribiendo superbloques y la información contable del sistema de ficheros: hecho
```

Montamos la partición

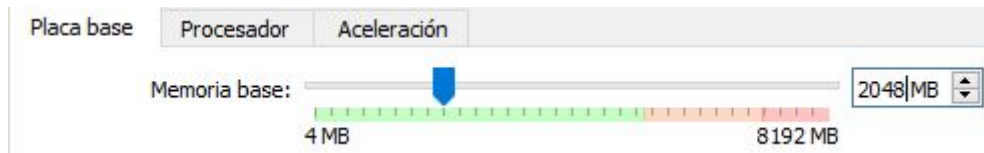
```
[root@localhost ~]# mkdir /mnt/discos
[root@localhost ~]# mount /dev/sdb1 /mnt/discos
[root@localhost ~]# mount /dev/sdb2 /mnt/discos
[root@localhost ~]# mount /dev/sdb3 /mnt/discos
[root@localhost ~]# █
```

Freebsd

1.- Modo de instalación del sistema operativo, cuanta ram necesita, que tamaño tiene el disco virtual, cuantas particiones tendrá. Que sistema de ficheros usa cada una de esas particiones.

Deben enumerarse todos los ítems que el sistema ha ido pidiendo cuando se ha ido instalando

En la instalación todo lo que me pregunto el menú de instalación lo deje predeterminado



Para este sistema pondremos 16 gb de almacenamiento interno reservado dinámicamente, luego le ponemos 2 gb de ram, tendrá solo 1 partición más las que se instalan predeterminadas. Para comprobar la particiones que tiene el sistema usare el comando `fdisk -l` y para saber que sistema de archivo tiene usare el comando `file -s` y la partición que queremos ver.

```
=>      0  33554368  ada0s1  BSD  (16G)
        0  31457280      1  freebsd-ufs  (15G)
31457280  1677312      2  freebsd-swap  (819M)
33134592  419776      - free -  (205M)
```

segun este comando la primera particion es freebsd-ufs que es la que se usa en este sistema, la segunda particion es la de swap y sobra 205 mb

2.- Mostrar el árbol de directorios de todos los sistemas operativos y anotar las coincidencias entre ellos.

Para mostrar el árbol de directorio usamos este comando:

```
ls -l /
```

```
total 32885
-rw-r--r--  2 root  wheel      951 Nov  1  2019 .cshrc
-rw-r--r--  2 root  wheel      470 Nov  1  2019 .profile
drwxrwxr-x  2 root  operator    512 Sep 29 11:52 .snap
-r-----  1 root  wheel 33554432 Sep 29 11:52 .su.journal
-r--r--r--  1 root  wheel    6181 Nov  1  2019 COPYRIGHT
drwxr-xr-x  2 root  wheel    1024 Nov  1  2019 bin
drwxr-xr-x  9 root  wheel    1536 Sep 29 11:55 boot
dr-xr-xr-x  9 root  wheel     512 Oct  1 13:24 dev
-rw-----  1 root  wheel   4096 Oct  1 13:24 entropy
drwxr-xr-x 25 root  wheel    2048 Oct  1 13:25 etc
drwxr-xr-x  5 root  wheel    1536 Nov  1  2019 lib
drwxr-xr-x  3 root  wheel     512 Sep 29 11:53 libexec
drwxr-xr-x  2 root  wheel     512 Nov  1  2019 media
drwxr-xr-x  2 root  wheel     512 Nov  1  2019 mnt
drwxr-xr-x  2 root  wheel     512 Nov  1  2019 net
dr-xr-xr-x  2 root  wheel     512 Nov  1  2019 proc
drwxr-xr-x  2 root  wheel   2560 Nov  1  2019 rescue
drwxr-xr-x  2 root  wheel     512 Nov  1  2019 root
drwxr-xr-x  2 root  wheel   2560 Nov  1  2019 sbin
lrwxr-xr-x  1 root  wheel      11 Nov  1  2019 sys -> usr/src/sys
drwxrwxrwt  6 root  wheel     512 Oct  1 13:24 tmp
drwxr-xr-x 14 root  wheel     512 Sep 29 11:55 usr
drwxr-xr-x 24 root  wheel     512 Oct  1 13:24 var
```

3.- Indicar el nivel de ejecución en el que cada sistema operativo entra por defecto. (Ejecutar en una terminal la orden runlevel)

No he encontrado la orden

4.-Indicar si el sistema operativo dispone de servidor gráfico. Indicar si dispone de escritorio y cual es su nombre.

Este sistema no tiene entorno gráfico, solo tiene una terminal.

5.- Agregar un usuario al sistema.

Para agregar un usuario usamos el comando:

adduser + nombre de usuario o useradd + nombre de usuario y configuramos su contraseña

```
Shell (sh csh tcsh nologin) [sh]:
Home directory [/home/g]:
Home directory permissions (Leave empty for default):
Use password-based authentication? [yes]: y
Use an empty password? (yes/no) [no]: n
Use a random password? (yes/no) [no]: n
Enter password:
Enter password again:
Lock out the account after creation? [no]: n
Username : g
Password : *****
Full Name :
Uid : 1001
Class :
Groups : g
Home : /home/g
Home Mode :
Shell : /bin/sh
Locked : no
OK? (yes/no): y
pw: invalid character 0xffff at position 0 in userid/group name
adduser: ERROR: There was an error adding user (g).
Add another user? (yes/no): n
Goodbye!
```

```
practicaLeo:*:1001:1001:User &:/home/practicaLeo:/bin/sh
```

Con lo cual, vemos que el usuario práctica está creado y tiene su propio directorio.

6.- Ver si tiene conectividad con el exterior (orden ping)

para comprobar si hay conexión a internet, hay que hacer por ejemplo una conexión con algo de fuera como por ejemplo www.google.es , usaremos este comando:

ping www.google.es, y si este comando nos da ping, significa que hay conexión a internet

```
root@usuario:~ # ping www.google.es
PING www.google.es (216.58.201.131): 56 data bytes
64 bytes from 216.58.201.131: icmp_seq=0 ttl=117 time=9.587 ms
64 bytes from 216.58.201.131: icmp_seq=1 ttl=117 time=10.654 ms
64 bytes from 216.58.201.131: icmp_seq=2 ttl=117 time=10.584 ms
^C
--- www.google.es ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 9.587/10.275/10.654/0.487 ms
root@usuario:~ #
```

```

root@usuario:~ # ifconfig
em0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
    options=81009b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,VLAN_HWCSUM,
FILTER>
    ether 08:00:27:ef:57:27
    inet6 fe80::a00:27ff:feef:5727%em0 prefixlen 64 scopeid 0x1
    inet 10.0.2.15 netmask 0xfffff00 broadcast 10.0.2.255
    media: Ethernet autoselect (1000baseT <full-duplex>)
    status: active
    nd6 options=23<PERFORMNUD,ACCEPT_RTADV,AUTO_LINKLOCAL>
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
    options=680003<RXCSUM,TXCSUM,LINKSTATE,RXCSUM_IPV6,TXCSUM_IPV6>
    inet6 ::1 prefixlen 128
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x2
    inet 127.0.0.1 netmask 0xff000000
    groups: lo
    nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>

```

7.- Mostrar la versión del núcleo en ejecución. Indicar donde se encuentra dicho núcleo en el sistema de ficheros de cada s.o.

Para mostrar la version de nucleo usamos el comando:

uname -a

```

root@usuario:~ # uname -a
FreeBSD usuario 12.1-RELEASE FreeBSD 12.1-RELEASE r354233 GENERIC amd64

```

```

root@usuario:~ # ls /boot
beastie.4th          frames.4th          logo-orbbw.4th
boot                gptboot            lua
boot0               gptzfsboot         mbr
boot0sio            isoboot            menu-commands.4th
boot1               kernel             menu.4th
boot1.efi           loader             menu.rc
boot1.efifat        loader.4th          menusets.4th
boot2               loader.conf         modules
brand-fbsd.4th      loader.efi          pmbr
brand.4th           loader.rc           pxeboot
cdboot              loader_4th          screen.4th
check-password.4th  loader_4th.efi     shortcuts.4th
color.4th           loader_lua          support.4th
defaults            loader_lua.efi     userboot.so
delay.4th           loader_simp         userboot_4th.so
device.hints        loader_simp.efi    userboot_lua.so
dtb                 logo-beastie.4th   version.4th
efi.4th             logo-beastiebw.4th zfs
entropy            logo-fbsdhw.4th    zfsboot
firmware            logo-orb.4th        zfsloader

```


8.-Agregar un disco duro de 10 gigas a dicho sistema. Crear tres particiones en él y montar las particiones de dicho disco en /mnt/discos. En la carpeta discos se creará una carpeta por cada partición. Antes de montar nada se habrá procedido a crear los sistemas de ficheros en cada partición. Debe documentarse sobre los sistemas de ficheros disponibles para cada S.O. De no existir /mnt/discos se crear

En este sistema operísticos usaremos gparted para hacer las particiones

Lo primero que haré es crear el disco ada1 y dentro del el las 3 particiones

```
root@usuario:~ # gpart create -s gpt ada1
ada1 created
root@usuario:~ # gpart show
=>      63  33554369  ada0  MBR   (16G)
      63          1      - free -   (512B)
      64  33554368      1  freebsd [active] (16G)

=>      0  33554368  ada0s1  BSD   (16G)
      0  31457280      1  freebsd-ufs (15G)
  31457280  1677312      2  freebsd-swap (819M)
  33134592   419776      - free -   (205M)

=>     40  20971440  ada1   GPT   (10G)
     40  20971440      - free -   (10G)

=>     40  20971440  diskid/DISK-UBc5d6a503-e4639eb5  GPT (10G)
     40  20971440      - free -   (10G)
```

En este caso cada partición será de 3 gb

```
root@usuario:~ # gpart add -t freebsd-ufs -s 3G  ada1
ada1p1 added
root@usuario:~ # gpart add -t freebsd-ufs -s 3G  ada1
ada1p2 added
root@usuario:~ # gpart add -t freebsd-ufs -s 3G  ada1
ada1p3 added
```

```
      40  20971440  ada1   GPT   (10G)
      40   6291456      1  freebsd-ufs (3.0G)
   6291496   6291456      2  freebsd-ufs (3.0G)
 12582952   6291456      3  freebsd-ufs (3.0G)
 18874408   2097072      - free -   (1.0G)
```

Luego con el comando newfs formatear las particiones y montarlas con el comando mount

```
root@usuario:~ # newfs /dev/ada1p1
/dev/ada1p1: 3072.0MB (6291456 sectors) block size 32768, fragment size 4096
        using 5 cylinder groups of 626.09MB, 20035 blks, 80256 inodes.
super-block backups (for fsck_ffs -b #) at:
    192, 1282432, 2564672, 3846912, 5129152
root@usuario:~ # newfs /dev/ada1p2
/dev/ada1p2: 3072.0MB (6291456 sectors) block size 32768, fragment size 4096
        using 5 cylinder groups of 626.09MB, 20035 blks, 80256 inodes.
super-block backups (for fsck_ffs -b #) at:
    192, 1282432, 2564672, 3846912, 5129152
root@usuario:~ # newfs /dev/ada1p3
/dev/ada1p3: 3072.0MB (6291456 sectors) block size 32768, fragment size 4096
        using 5 cylinder groups of 626.09MB, 20035 blks, 80256 inodes.
super-block backups (for fsck_ffs -b #) at:
    192, 1282432, 2564672, 3846912, 5129152
root@usuario:~ #
```

```
ada1p1n ada1p2n ada1p3n
root@usuario:~ # mount /dev/ada1p1 /mnt/discos
root@usuario:~ # mount /dev/ada1p2 /mnt/discos
root@usuario:~ # mount /dev/ada1p3 /mnt/discos
root@usuario:~ #
```

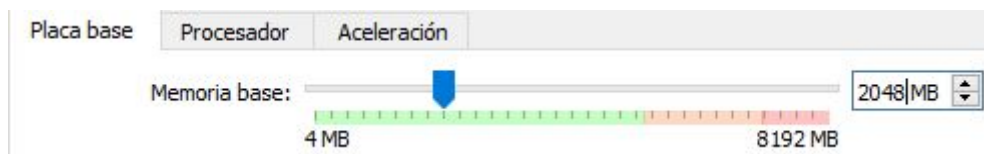
```
root@usuario:~ # mount /dev/ada1p1 /mnt/discos
root@usuario:~ # mount /dev/ada1p2 /mnt/discos
root@usuario:~ # mount /dev/ada1p3 /mnt/discos
root@usuario:~ #
```

Openbsd

1.- Modo de instalación del sistema operativo, cuanta ram necesita, que tamaño tiene el disco virtual, cuantas particiones tendrá. Que sistema de ficheros usa cada una de esas particiones.

Deben enumerarse todos los ítems que el sistema ha ido pidiendo cuando se ha ido instalando

En la instalación todo lo que me pregunto el menú de instalación lo deje predeterminado



Para este sistema pondremos 16 gb de almacenamiento interno reservado dinámicamente, luego le ponemos 2 gb de ram, tendrá solo 1 partición más las que se instalan predeterminadas. Para comprobar la particiones que tiene el sistema usare el comando `fdisk -l` y para saber que sistema de archivo tiene usare el comando `file -s` y la partición que queremos ver.

2.- Mostrar el árbol de directorios de todos los sistemas operativos y anotar las coincidencias entre ellos.

Para mostrar el árbol de directorio usamos este comando:

`ls -l /`

```
leo# ls -l /
total 93032
-rw-r--r--  1 root  wheel      578 May  7 18:51 .cshrc
-rw-r--r--  1 root  wheel      468 May  7 18:51 .profile
drwxr-xr-x  2 root  wheel      512 May  7 18:51 altroot
drwxr-xr-x  2 root  wheel     1024 May  7 18:52 bin
-rw-r--r--  1 root  wheel    89584 Oct  3 12:11 boot
-rwx-----  1 root  wheel  18539288 Oct  3 14:14 bsd
-rwx-----  1 root  wheel  18544456 Oct  3 12:11 bsd.booted
-rw-----  1 root  wheel  10352464 Oct  3 14:09 bsd.rd
drwxr-xr-x  4 root  wheel     19456 Oct  3 14:13 dev
drwxr-xr-x 23 root  wheel     1536 Oct  3 14:14 etc
drwxr-xr-x  2 root  wheel      512 May  7 18:51 home
drwxr-xr-x  2 root  wheel      512 May  7 18:51 mnt
drwx-----  3 root  wheel      512 Oct  3 14:10 root
drwxr-xr-x  2 root  wheel     1536 May  7 18:52 sbin
lrwxrwx---  1 root  wheel       11 May  7 18:51 sys -> usr/src/sys
drwxrwxrwt  6 root  wheel      512 Oct  3 14:14 tmp
drwxr-xr-x 16 root  wheel      512 Oct  3 12:11 usr
drwxr-xr-x 23 root  wheel      512 May  7 19:50 var
leo#
```

3.- Indicar el nivel de ejecución en el que cada sistema operativo entra por defecto. (Ejecutar en una terminal la orden `runlevel`)

No he encontrado la orden de `runlevel` en `openbsd`

4.-Indicar si el sistema operativo dispone de servidor gráfico. Indicar si dispone de escritorio y cual es su nombre.

Este sistema no tiene entorno gráfico, solo tiene una terminal.

5.- Agregar un usuario al sistema.

Para agregar un usuario usamos el comando:

`adduser + nombre de usuario` o `useradd + nombre de usuario` y configuramos su contraseña

```
Enter shell csh ksh nologin sh [ksh]:
Uid [1001]:
Login group Practica [Practica]:
Login group is ``Practica``. Invite Practica into other groups: guest no
[nol]:
Login class authpf bgpd daemon default pbuild staff unbound
[default]:
Enter password []:
Enter password again []:

Name:      Practica
Password:   ****
Fullname:   Practica S0
Uid:       1001
Gid:       1001 (Practica)
Groups:     Practica
Login Class: default
HOME:      /home/Practica
Shell:     /bin/ksh
OK? (y/n) [y]: y
Added user ``Practica``
Copy files from /etc/skel to /home/Practica
Add another user? (y/n) [y]: n
Goodbye!
leo# ca
```

Y para comprobar que el usuario se ha creado correctamente hay que usar este comando:

`cat /etc/passwd`

```
Practica:*:1001:1001:Practica S0:/home/Practica:/bin/ksh
leo#
```

Con lo cual, vemos que el usuario práctica está creado y tiene su propio directorio.

6.- Ver si tiene conectividad con el exterior (orden ping)

para comprobar si hay conexión a internet, hay que hacer por ejemplo una conexión con algo de fuera como por ejemplo `www.google.es` , usaremos este comando:

`ping www.google.es`, y si este comando nos da ping, significa que hay conexión a internet

```

PING www.google.es (216.58.201.131): 56 data bytes
64 bytes from 216.58.201.131: icmp_seq=0 ttl=117 time=9.918 ms
64 bytes from 216.58.201.131: icmp_seq=1 ttl=117 time=9.148 ms
64 bytes from 216.58.201.131: icmp_seq=2 ttl=117 time=9.595 ms
^C
--- www.google.es ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/std-dev = 9.148/9.553/9.918/0.316 ms
leo#

```

```

leo# ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 32768
    index 3 priority 0 llprio 3
    groups: lo
    inet6 ::1 prefixlen 128
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x3
    inet 127.0.0.1 netmask 0xff000000
em0: flags=808843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST,AUTOCONF4> mtu 1500
    lladdr 08:00:27:d9:75:49
    index 1 priority 0 llprio 3
    groups: egress
    media: Ethernet autoselect (1000baseT full-duplex)
    status: active
    inet 10.0.2.15 netmask 0xfffff00 broadcast 10.0.2.255
enc0: flags=0<>
    index 2 priority 0 llprio 3
    groups: enc
    status: active
pflog0: flags=141<UP,RUNNING,PROMISC> mtu 33136
    index 4 priority 0 llprio 3
    groups: pflog
leo#

```

7.- Mostrar la versión del núcleo en ejecución. Indicar donde se encuentra dicho núcleo en el sistema de ficheros de cada s.o.

```

leo# uname -a
OpenBSD leo.my.domain 6.7 GENERIC#179 amd64
leo#

```

8.-Agregar un disco duro de 10 gigas a dicho sistema. Crear tres particiones en él y montar las particiones de dicho disco en /mnt/discos. En la carpeta discos se creará una carpeta por cada partición. Antes de montar nada se habrá procedido a crear los sistemas de ficheros en cada partición. Debe documentarse sobre los sistemas de ficheros disponibles para cada S.O. De no existir /mnt/discos se crear

No he sido capaz de encontrar los comandos, ya que con gpart no funcionaba y tampoco con fdisk

Minix

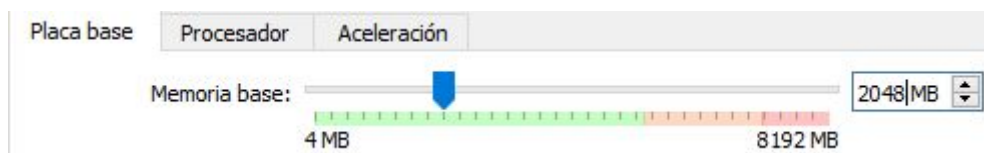
Ya que el minix no funcionaba de ninguna manera ya que la máquina se quedaba pillada una vez instalada, instale el Debian y voy a hacer las mismas cosas. En el video sale el error que tengo

Debian

1.- Modo de instalación del sistema operativo, cuanta ram necesita, que tamaño tiene el disco virtual, cuantas particiones tendrá. Que sistema de ficheros usa cada una de esas particiones.

Deben enumerarse todos los ítems que el sistema ha ido pidiendo cuando se ha ido instalando

En la instalación todo lo que me pregunto el menú de instalación lo deje predeterminado



Para este sistema pondremos 16 gb de almacenamiento interno reservado dinámicamente, luego le ponemos 2 gb de ram, tendrá solo 1 partición más las que se instalan predeterminadas. Para comprobar la particiones que tiene el sistema usará el comando `fdisk -l` y para saber que sistema de archivo tiene usare el comando `file -s` y la partición que queremos ver.

```
root@leo:/home/leo# fdisk -l
Disco /dev/sda: 16 GiB, 17179869184 bytes, 33554432 sectores
Modelo de disco: VBOX HARDDISK
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0x03bbe73b
```

Disposit.	Inicio	Comienzo	Final	Sectores	Tamaño	Id	Tipo
/dev/sda1	*	2048	31457279	31455232	15G	83	Linux
/dev/sda2		31459326	33552383	2093058	1022M	5	Extendida
/dev/sda5		31459328	33552383	2093056	1022M	82	Linux swap / Solaris


```

root@leo:/home/leo# file -s /dev/sda1
/dev/sda1: Linux rev 1.0 ext4 filesystem data, UUID=a980b9c2-c778-4c20-bb86-c2a5
acfedae2 (needs journal recovery) (extents) (64bit) (large files) (huge files)
root@leo:/home/leo# file -s /dev/sda2
/dev/sda2: DOS/MBR boot sector; partition 1 : ID=0x82, start-CHS (0x3ff,254,63),
end-CHS (0x3ff,254,63), startsector 2, 2093056 sectors, extended partition tabl
e (last)
root@leo:/home/leo# file -s /dev/sda5
/dev/sda5: Linux/i386 swap file (new style), version 1 (4K pages), size 261631 p
ages, no label, UUID=7d0160f9-44f8-4e64-81ce-db258318b039
root@leo:/home/leo# █

```

2.- Mostrar el árbol de directorios de todos los sistemas operativos y anotar las coincidencias entre ellos.

Para mostrar el árbol de directorio usamos este comando:

`ls -l /`

```

root@leo:/home/leo# ls -l /
total 60
lrwxrwxrwx    1 root root      7 oct 13 09:42 bin -> usr/bin
drwxr-xr-x    3 root root  4096 oct 13 09:58 boot
drwxr-xr-x   17 root root  3160 oct 13 09:59 dev
drwxr-xr-x  119 root root  4096 oct 13 10:13 etc
drwxr-xr-x    3 root root  4096 oct 13 09:58 home
lrwxrwxrwx    1 root root    31 oct 13 09:44 initrd.img -> boot/initrd.img-4.19.0-11-amd
64
lrwxrwxrwx    1 root root    31 oct 13 09:44 initrd.img.old -> boot/initrd.img-4.19.0-11
-amd64
lrwxrwxrwx    1 root root      7 oct 13 09:42 lib -> usr/lib
lrwxrwxrwx    1 root root      9 oct 13 09:42 lib32 -> usr/lib32
lrwxrwxrwx    1 root root      9 oct 13 09:42 lib64 -> usr/lib64
lrwxrwxrwx    1 root root     10 oct 13 09:42 libx32 -> usr/libx32
drwx-----   2 root root 16384 oct 13 09:42 lost+found
drwxr-xr-x    3 root root  4096 oct 13 09:42 media
drwxr-xr-x    2 root root  4096 oct 13 09:43 mnt
drwxr-xr-x    2 root root  4096 oct 13 09:43 opt
dr-xr-xr-x  146 root root      0 oct 13 2020 proc
drwx-----   4 root root  4096 oct 13 10:11 root
drwxr-xr-x   23 root root   620 oct 13 10:00 run
lrwxrwxrwx    1 root root      8 oct 13 09:42/sbin -> usr/sbin
drwxr-xr-x    2 root root  4096 oct 13 09:43 srv
dr-xr-xr-x   13 root root      0 oct 13 09:59 sys
drwxrwxrwt   15 root root  4096 oct 13 10:15 tmp
drwxr-xr-x   14 root root  4096 oct 13 09:52 usr
drwxr-xr-x   11 root root  4096 oct 13 09:43 var
lrwxrwxrwx    1 root root     28 oct 13 09:44 vmlinuz -> boot/vmlinuz-4.19.0-11-amd64
lrwxrwxrwx    1 root root     28 oct 13 09:44 vmlinuz.old -> boot/vmlinuz-4.19.0-11-amd64
root@leo:/home/leo#

```

3.- Indicar el nivel de ejecución en el que cada sistema operativo entra por defecto.
(Ejecutar en una terminal la orden runlevel)

```
root@leo:/home/leo# runlevel
N 5
```

4.- Indicar si el sistema operativo dispone de servidor gráfico. Indicar si dispone de escritorio y cual es su nombre.

Este sistema es de entorno gráfico y dispone de escritorio. Para saber que tipo o que nombre de entorno gráfico se usa el comando:

ls /usr/bin/*session

```
root@leo:/home/leo# ls /usr/bin/*session
/usr/bin/dbus-run-session  /usr/bin/gnome-session-custom-session
/usr/bin/gnome-session
```

En este caso es un escritorio gnome

5.- Agregar un usuario al sistema.

Para agregar un usuario usamos el comando:

adduser + nombre de usuario o useradd + nombre de usuario y configuramos su contraseña

```
root@leo:/home/leo# adduser practicaleo
Añadiendo el usuario `practicaleo' ...
Añadiendo el nuevo grupo `practicaleo' (1001) ...
Añadiendo el nuevo usuario `practicaleo' (1001) con grupo `practicaleo' ...
Creando el directorio personal `/home/practicaleo' ...
Copiando los ficheros desde `/etc/skel' ...
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
Cambiando la información de usuario para practicaleo
Introduzca el nuevo valor, o pulse INTRO para usar el valor predeterminado
    Nombre completo []:
    Número de habitación []:
    Teléfono del trabajo []:
    Teléfono de casa []:
    Otro []:
¿Es correcta la información? [S/n] s
```

Y para comprobar que el usuario se ha creado correctamente hay que usar este comando:

`cat /etc/passwd`

```
practicaleo:x:1001:1001:,,,:/home/practicaleo:/bin/bash
```

Con lo cual, vemos que el usuario practicaleo está creado y tiene su propio directorio.

6.- Ver si tiene conectividad con el exterior (orden ping)

para comprobar si hay conexión a internet, hay que hacer por ejemplo una conexión con algo de fuera como por ejemplo www.google.es , usaremos este comando:

ping www.google.es, y si este comando nos da ping, significa que hay conexión a internet

```
root@leo:/home/leo# ping www.google.es
PING www.google.es (216.58.201.131) 56(84) bytes of data.
64 bytes from mad06s25-in-f131.1e100.net (216.58.201.131): icmp_seq=1 ttl=117 time=9.10
0 ms
64 bytes from mad06s25-in-f131.1e100.net (216.58.201.131): icmp_seq=2 ttl=117 time=10.7
ms
64 bytes from mad06s25-in-f131.1e100.net (216.58.201.131): icmp_seq=3 ttl=117 time=9.94
ms
^C
--- www.google.es ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 6ms
rtt min/avg/max/mdev = 9.944/10.207/10.681/0.345 ms
```

```
root@leo:/home/leo# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe2c:960c prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:2c:96:0c txqueuelen 1000 (Ethernet)
    RX packets 1935 bytes 2476900 (2.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 736 bytes 55683 (54.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 20 bytes 1116 (1.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 20 bytes 1116 (1.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

7.- Mostrar la versión del núcleo en ejecución. Indicar dónde se encuentra dicho núcleo en el sistema de ficheros de cada s.o.

```
root@leo:/home/leo# uname -a
Linux leo 4.19.0-11-amd64 #1 SMP Debian 4.19.146-1 (2020-09-17) x86_64 GNU/Linux
```

```
root@leo:/home/leo# ls -l /
total 60
lrwxrwxrwx   1 root root    7 oct 13 09:42 bin -> usr/bin
drwxr-xr-x   3 root root 4096 oct 13 09:58 boot
drwxr-xr-x  17 root root 3160 oct 13 09:59 dev
drwxr-xr-x 119 root root 4096 oct 13 10:21 etc
drwxr-xr-x   4 root root 4096 oct 13 10:20 home
lrwxrwxrwx   1 root root   31 oct 13 09:44 initrd.img -> boot/initrd.img-4.19.0-11-amd64
lrwxrwxrwx   1 root root   31 oct 13 09:44 initrd.img.old -> boot/initrd.img-4.19.0-11-amd64
lrwxrwxrwx   1 root root    7 oct 13 09:42 lib -> usr/lib
lrwxrwxrwx   1 root root    9 oct 13 09:42 lib32 -> usr/lib32
lrwxrwxrwx   1 root root    9 oct 13 09:42 lib64 -> usr/lib64
lrwxrwxrwx   1 root root   10 oct 13 09:42 libx32 -> usr/libx32
drwx-----  2 root root 16384 oct 13 09:42 lost+found
drwxr-xr-x   3 root root 4096 oct 13 09:42 media
drwxr-xr-x   2 root root 4096 oct 13 09:43 mnt
drwxr-xr-x   2 root root 4096 oct 13 09:43 opt
dr-xr-xr-x 147 root root    0 oct 13 10:19 proc
drwx-----  4 root root 4096 oct 13 10:11 root
drwxr-xr-x  23 root root  620 oct 13 10:00 run
lrwxrwxrwx   1 root root    8 oct 13 09:42 sbin -> usr/sbin
drwxr-xr-x   2 root root 4096 oct 13 09:43 srv
dr-xr-xr-x  13 root root    0 oct 13 10:17 sys
drwxrwxrwt  15 root root 4096 oct 13 10:24 tmp
drwxr-xr-x  14 root root 4096 oct 13 09:52 usr
drwxr-xr-x  11 root root 4096 oct 13 09:43 var
lrwxrwxrwx   1 root root   28 oct 13 09:44 vmlinuz -> boot/vmlinuz-4.19.0-11-amd64
lrwxrwxrwx   1 root root   28 oct 13 09:44 vmlinuz.old -> boot/vmlinuz-4.19.0-11-amd64
```

8.-Agregar un disco duro de 10 gigas a dicho sistema. Crear tres particiones en él y montar las particiones de dicho disco en /mnt/discos. En la carpeta discos se creará una carpeta por cada partición. Antes de montar nada se habrá procedido a crear los sistemas de ficheros en cada partición. Debe documentarse sobre los sistemas de ficheros disponibles para cada S.O. De no existir /mnt/discos se crear

A los discos primeros les puse 300.000 y a el ultimo le puse lo que sobraba

A continuación formatear las particiones y poder xt4 con el comando mkfs.ext4 /dev/sdb para cada 1 de las tres particiones

Disposit.	Inicio	Comienzo	Final	Sectores	Tamaño	Id	Tipo
/dev/sdb1		2048	300000	297953	145,5M	83	Linux
/dev/sdb2		301056	600000	298945	146M	83	Linux
/dev/sdb3		600064	20971519	20371456	9,7G	83	Linux

con el comando /sbin/mkfs.ext4 les doy formato a las particiones

```
leo@leo:~$ sudo /sbin/mkfs.ext4 /dev/sdb1
mke2fs 1.44.5 (15-Dec-2018)
Creating filesystem with 148976 1k blocks and 37392 inodes
Filesystem UUID: 969435ef-e8ab-4b48-a8fc-026c4cc2326a
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

leo@leo:~$ sudo /sbin/mkfs.ext4 /dev/sdb2
mke2fs 1.44.5 (15-Dec-2018)
Creating filesystem with 149472 1k blocks and 37392 inodes
Filesystem UUID: ff6d97ff-89fd-454d-b45f-34d2f389741c
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
```

```
root@leo:/home/leo# mount /dev/sdb1 /mnt/discos/
root@leo:/home/leo# mount /dev/sdb2 /mnt/discos/
root@leo:/home/leo# mount /dev/sdb3 /mnt/discos/
root@leo:/home/leo#
```
