

GESTIÓN AVANZADA DE PARTICIONES

Pol Muel Garcia
Funaments de Maquinari
8/1/2025

1. INTRODUCCIÓN.....	2
2. Configuración del disco virtual.....	3
3. Creación de particiones.....	4
4. Preparación y Formateo.....	4
5. Montaje y desmontaje.....	6
6. Automontaje al sistema.....	7
7. Análisis de rendimiento.....	8
8. Conclusiones.....	9

1. INTRODUCCIÓN

En esta práctica vamos a aprender como particionar y aplicar diferentes sistemas de archivos según la conveniencia en cada partición hecha. Además, veremos como montar y desmontar las particiones utilizando la terminal, al igual que los demás puntos anteriormente comentados.

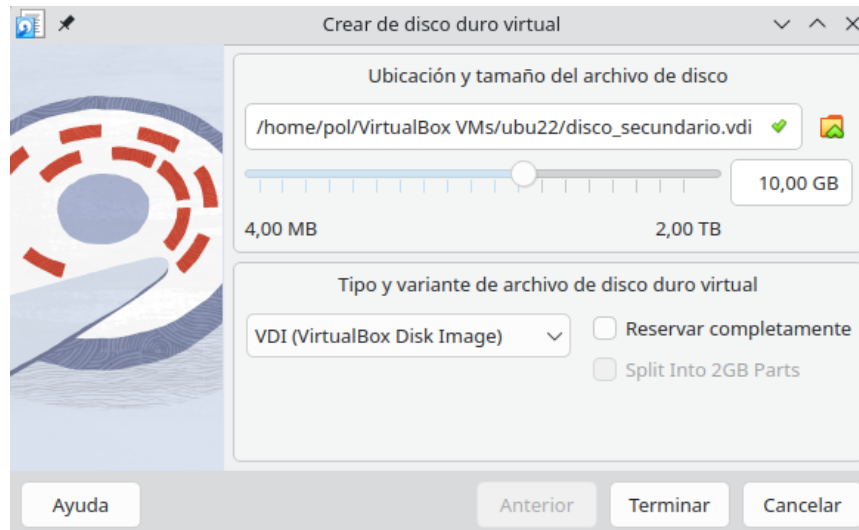
Más adelante configuraremos un automontaje de particiones al iniciar el ordenador, para así poder configurar donde se pondrán automáticamente al iniciar el sistema. Y una vez montadas, compararemos el rendimiento entre las diferentes particiones con sus pertinentes sistemas de archivos para ver cuál rinde mejor para así tener una clara idea de que utilizar si buscamos optimización y rapidez.

Por lo que antes de empezar, necesitaremos una máquina virtual con Ubuntu 22.04 ya configurada e instalada, también tendremos que incorporarle un disco duro secundario que tendrá 10 GB.

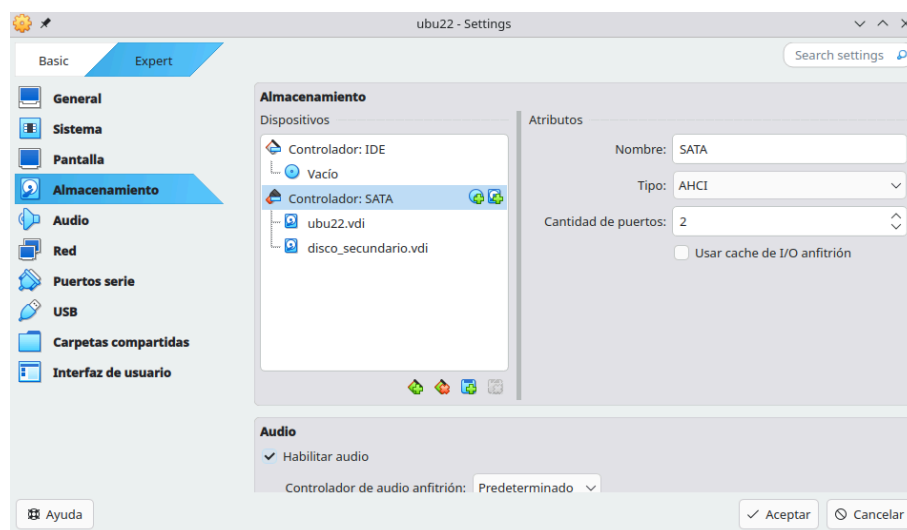
2. Configuración del disco virtual

Para empezar la práctica, tendremos que crear un disco virtual y añadirlo en la máquina virtual de Ubuntu 22.

Este disco tendrá una capacidad de 10Gb, y lo nombraremos disco_secundario.vdi



Una vez creado, lo añadimos en la máquina virtual desde su menú de configuración.



Ahora deberíamos tener dos discos en la máquina, por lo que vamos a iniciarla para poder comprobar que se detectan correctamente con el comando lsblk.

```
sda      8:0    0    25G    0 disk
├─sda1    8:1    0     1M    0 part
├─sda2    8:2    0   513M    0 part /boot/efi
└─sda3    8:3    0   24,5G    0 part /
sdb      8:16   0    10G    0 disk
```

3. Creación de particiones

En la siguiente etapa de la práctica, vamos a hacer particiones en el disco secundario que añadimos en el apartado anterior. Por lo que esto vamos a hacerlo con el comando fdisk.

```
Bienvenido a fdisk (util-linux 2.37.2).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

El dispositivo no contiene una tabla de particiones reconocida.
Se ha creado una nueva etiqueta de disco DOS con el identificador de disco 0x096b6324.

Orden (m para obtener ayuda): n
Tipo de partición
  p  primaria (0 primary, 0 extended, 4 free)
  e  extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): p
Número de partición (1-4, valor predeterminado 1):
Primer sector (2048-20971519, valor predeterminado 2048):
Last sector, +/-sectors or +/-size[K,M,G,T,P] (2048-20971519, valor predeterminado 20971519): +4
G

Crea una nueva partición 1 de tipo 'Linux' y de tamaño 4 GiB.
Terminal para obtener ayuda): n
Tipo de partición
  p  primaria (1 primary, 0 extended, 3 free)
  e  extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): p
Número de partición (2-4, valor predeterminado 2):
Primer sector (8390656-20971519, valor predeterminado 8390656):
Last sector, +/-sectors or +/-size[K,M,G,T,P] (8390656-20971519, valor predeterminado 20971519):
+4G

Crea una nueva partición 2 de tipo 'Linux' y de tamaño 4 GiB.
Orden (m para obtener ayuda):
```

4. Preparación y Formateo

A continuación, como nos pide la práctica, vamos a cambiar el sistema de archivos de las particiones, donde la primera partición será un sistema de archivos ext4, y la segunda será ntfs.

Para ello utilizaremos estos dos comandos para hacerlo:

- `sudo mkfs.ext4 /dev/sdb1` → Para hacer la partición ext4

```
muelas@muelas-VirtualBox:~$ sudo mkfs.ext4 /dev/sdb1
mke2fs 1.46.5 (30-Dec-2021)
Se está creando un sistema de ficheros con 1048576 bloques de 4k y 262144 nodos-i
UUID del sistema de ficheros: c20523f8-e67b-41e7-94cd-b95f85bd07b0
Respalos del superbloque guardados en los bloques:
 32768, 98304, 163840, 229376, 294912, 819200, 884736

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: hecho
```

- `sudo mkfs.ntfs /dev/sdb2` → Para hacer la partición ntfs

```
muelas@muelas-VirtualBox:~$ sudo mkfs.ntfs /dev/sdb2
Cluster size has been automatically set to 4096 bytes.
Initializing device with zeroes: 100% - Done.
Creating NTFS volume structures.
mkntfs completed successfully. Have a nice day.
muelas@muelas-VirtualBox:~$
```

Para finalizar el apartado vamos a utilizar lo restante de la memoria en el segundo disco para hacer una partición SWAP. Estas particiones actúan como memoria de intercambio, y se usa como si fuera memoria RAM cuando esta no es suficiente

Por lo que, empezaremos creando, al igual que antes, la partición 3, y la modificaremos la tecla t (dentro de fdisk) para poder asignar el sistema de archivos en swap, al pedirnos un hexalias, este sistema es el número 82.

Dispositivo	Inicio	Comienzo	Final	Sectores	Tamaño	Id	Tipo
/dev/sdb1		2048	8390655	8388608	4G	83	Linux
/dev/sdb2		8390656	16779263	8388608	4G	83	Linux
/dev/sdb3		16779264	20971519	4192256	2G	82	Linux swap / Solaris

A continuación haremos una serie de comandos para crear él un archivo swap y activarlo para su funcionamiento, lo cual haremos con estos comandos:

```
muelas@muelas-VirtualBox:~$ sudo mkswap /dev/sdb3
Configurando espacio de intercambio versión 1, tamaño = 2 GiB (2146430976 bytes)
sin etiqueta, UUID=d126f7ff-86d4-42db-b340-bac682d693f0
muelas@muelas-VirtualBox:~$ sudo swapon /dev/sdb3
```

Para comprobar que se ha hecho todo este proceso correctamente, utilizaremos el comando lsblk -f, donde nos mostrara los discos y sus particiones con algo más de información gracias al -f.

```
sdb
├─sdb1
│   └─ext4 1.0 c20523f8-e67b-41e7-94cd-b95f85bd07b0
├─sdb2
│   └─ntfs 2D8CA0C43FE98688
└─sdb3
    └─swap 1 d126f7ff-86d4-42db-b340-bac682d693f0 [SWAP]
```

5. Montaje y desmontaje

Ahora tenemos las particiones creadas, solo nos queda montarlas en el punto de nuestro equipo que queramos, lo haremos con el comando “sudo mount /dev/sdb? /mnt/directorio”.

Pero primeramente, deberemos crear las carpetas donde vamos a montar las particiones, lo cual haremos con el comando “sudo mkdir nombre_carpetas”.

```
muelas@muelas-VirtualBox:/mnt$ sudo mkdir part1
muelas@muelas-VirtualBox:/mnt$ sudo mkdir part2
muelas@muelas-VirtualBox:/mnt$ sudo mkdir part3
```

Una vez creadas ya podremos montar las particiones con el comando anteriormente comentado.

```
muelas@muelas-VirtualBox:/mnt$ sudo mount /dev/sdb1 /mnt/part1
muelas@muelas-VirtualBox:/mnt$ sudo mount /dev/sdb2 /mnt/part2
muelas@muelas-VirtualBox:/mnt$ sudo mount /dev/sdb3 /mnt/part3
mount: /mnt/part3: tipo de sistema de ficheros 'swap' desconocido.
```

Como nos dice la terminal, la tercera partición no vamos a poder montarla, ya que esta es una partición swap. Para confirmar que se ha montado introduciremos lsblk en terminal.

```
┌-sdb1   8:17   0    4G  0 part /mnt/part1
├-sdb2   8:18   0    4G  0 part /mnt/part2
└-sdb3   8:19   0    2G  0 part
sr0      11:0    1 1024M  0 rom
muelas@muelas-VirtualBox:/mnt$
```

Una vez comprobado vamos a desmontar las particiones para poder seguir con el próximo paso de la práctica, las desmontaremos con el comando “sudo umount /mnt/directorio”

```
muelas@muelas-VirtualBox:/mnt$ sudo umount /mnt/part1
muelas@muelas-VirtualBox:/mnt$ sudo umount /mnt/part2
muelas@muelas-VirtualBox:/mnt$
```

6. Automontaje al sistema

Para automatizar el montaje automático al iniciar, debemos ir al archivo que se encuentra en /etc y modificar su contenido, por lo que introduciremos “sudo nano /etc/fstab”.

```
muelas@muelas-VirtualBox: /mnt
GNU nano 6.2 /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda3 during installation
UUID=b46fc4d9-0059-41bd-889a-1e05345dfe5f / ext4 errors=remoun
# /boot/efi was on /dev/sda2 during installation
UUID=96C3-A88E /boot/efi vfat umask=0077 0 1
/swapfile none swap none 0 sw

/dev/sdb1 /mnt/part1 ext4 defaults 0 2
/dev/sdb2 /mnt/part2 ntfs defaults 0 2
/dev/sdb3 none swap sw 0 0
```

Dentro del archivo introduciremos la dirección de la partición, en mi caso /dev/sdb?, donde queremos que se monte (/mnt/part1), el sistema de archivos que utiliza, en este caso ext4 y ntfs, y distintos parámetros que suelen ser los que se ven en pantalla, defaults 0 2, guardamos y salimos.

Para verificar que funciona, pondremos el comando “sudo mount -a”, y a continuación vemos las particiones con “lsblk”.

```
muelas@muelas-VirtualBox:/mnt$ sudo mount -a
muelas@muelas-VirtualBox:/mnt$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0       7:0    0   4K  1 loop /snap/bare/5
loop1       7:1    0  61,9M 1 loop /snap/core20/1376
loop2       7:2    0  63,7M 1 loop /snap/core20/2434
loop3       7:3    0 155,4M 1 loop /snap/firefox/1154
loop4       7:4    0 349,7M 1 loop /snap/gnome-3-38-2004/143
loop5       7:5    0 248,8M 1 loop /snap/gnome-3-38-2004/99
loop6       7:6    0  79,1M 1 loop /snap/gtk-common-themes/1531
loop7       7:7    0  91,7M 1 loop /snap/gtk-common-themes/1535
loop8       7:8    0  45,9M 1 loop /snap/snap-store/575
loop9       7:9    0  43,6M 1 loop /snap/snapd/15177
loop10      7:10   0   288K 1 loop /snap/snapd-desktop-integration/7
sda         8:0    0   25G  0 disk
├─sda1      8:1    0    1M  0 part
├─sda2      8:2    0   513M  0 part /boot/efi
└─sda3      8:3    0   24,5G  0 part /
sdb         8:16   0   10G  0 disk
├─sdb1      8:17   0    4G  0 part /mnt/part1
├─sdb2      8:18   0    4G  0 part /mnt/part2
└─sdb3      8:19   0    2G  0 part
sr0        11:0    1 1024M  0 rom
```


7. Análisis de rendimiento

Para evaluar las particiones, realizaremos un análisis de rendimiento, así podremos comprobar su efectividad y funcionamiento.

Primero miraremos la escritura y lectura de la primera partición que tiene un sistema de archivos ext4.

```
muelas@muelas-VirtualBox:/mnt$ sudo dd count=100 bs=1M if=/dev/sdb1 of=/home/ran  
dom_100MB.out  
100+0 registros leídos  
100+0 registros escritos  
104857600 bytes (105 MB, 100 MiB) copied, 0,0686304 s, 1,5 GB/s  
muelas@muelas-VirtualBox:/mnt$
```

Y a continuación la segunda partición con un sistema de archivos ntfs.

```
muelas@muelas-VirtualBox:/mnt$ sudo dd count=100 bs=1M if=/dev/sdb2 of=/home/ran  
dom_100MB.out  
100+0 registros leídos  
100+0 registros escritos  
104857600 bytes (105 MB, 100 MiB) copied, 0,359213 s, 292 MB/s  
muelas@muelas-VirtualBox:/mnt$
```

Por lo que podemos apreciar en las imágenes, la segunda partición tiene un rendimiento mucho más lento tanto de lectura como de escritura, ya que hay diferencias apreciables, como 0,3s más y 1,2GB menos.

Entonces podemos dar el veredicto de que el sistema de archivos ext4 tiene un rendimiento mucho mayor, al menos en el sistema operativo Ubuntu 22.

8. Conclusiones

Hemos podido observar varias grandes enseñanzas en esta práctica como algo tan simple, pero al igual que importante qué simple, crear particiones, porque están tienen un uso muy importante si se utiliza el ordenador de manera profesional y se quiere dar un buen uso.

Además, en caso de querer eliminar alguna partición o formatearla, con esta, guía paso a paso, se puede consultar para una posible resolución de conflictos. También se ha aprendido a dar un sistema de archivos a una partición, y que si se utiliza el Ubuntu 22, es mucho mejor escoger un sistema ext4 antes que ntfs para poder disfrutar de un buen rendimiento, y por consecuencia, una buena resolución.