

Raid 1 en Linux

Arrancamos VirtualBox

Arrancamos la máquina virtual que contenga la distro de Ubuntu

Creamos dos discos duros de 100 megas.

Comprobamos que los **discos duros** añadidos se encuentran ahí.

```
root@kali:~# fdisk -l
Disk /dev/sdc: 102 MiB, 106954752 bytes, 208896 sectors
Disk model: VMware Virtual S
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/sdb: 102 MiB, 106954752 bytes, 208896 sectors
Disk model: VMware Virtual S
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VMware Virtual S
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: 0xe8c0201e
```

Instalamos el paquete **mdadm**, que gestiona los dispositivos RAID por software.

```
# apt-get install mdadm
```

Creamos el RAID 1 con el comando:

```
# mdadm -C /dev/md0 --level=1 --raid-devices=2 /dev/sdb /dev/sdc
```

Donde /dev/md0 es el nombre del nuevo dispositivo RAID, level=1 indica un RAID 1, raid-devices=2 indica que son dos discos.

```

root@kali:~# mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/sdb /dev/sdc
mdadm: Note: this array has metadata at the start and
may not be suitable as a boot device. If you plan to
store '/boot' on this device please ensure that
your boot-loader understands md/v1.x metadata, or use
--metadata=0.90
Continue creating array? yes
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.

```

Comprobamos con **fdisk-l** que el nuevo dispositivo está **disponible** y tiene 100 MB, como los originales.

```

Disk /dev/md0: 101 MiB, 105906176 bytes, 206848 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

```

Ya **podemos** **trabajar** con él como un disco cualquiera. **crear** **particiones** con **fdisk**, **formatearlo** con **mkfs**, etc. Vamos a crear una partición, la formatearemos y meteremos un fichero de 50 MB. Para crear la partición usamos:

fdisk /dev/md0

Este comando nos ofrece un menú con múltiples operaciones. **Elegimos n** para **crear** la nueva **partición** y en las **opciones** dejamos los valores **por defecto** (tipo p [primaria], número 1 y todos los sectores disponibles). Finalmente ejecutamos w para escribir los cambios al disco.

```

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): 1
First sector (2048-206847, default 2048): 2048
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-206847, default 206847): 206847
Created a new partition 1 of type 'Linux' and of size 100 MiB.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

```

Si aquí damos a intro sin escribir 2048, coge 2048 por defecto

Si aquí ticamos intro sin escribir 206847, pondrá 206847 por defecto

Si hacemos **fdisk -l** podemos ver el nuevo nombre asignado, es decir al anterior **md0**, se le suma **p1**, **p** (**primaria**) y **1** (**número de partición**)

| Device | Boot | Start | End | Sectors | Size | Id | Type |
|------------|------|-------|--------|---------|------|----|-------|
| /dev/md0p1 | | 2048 | 206847 | 204800 | 100M | 83 | Linux |

Formatearemos la partición y la montaremos:

mkfs /dev/md0p1 (formateamos)

mkdir /mnt/raid1 (creamos el directorio donde montaremos la nueva unidad)

```
# mount /dev/md0p1 /mnt/raid1 (montamos la unidad en el directorio creado)
```

Con estos tres comandos creamos el sistema de ficheros y lo montamos para empezar a usarlo.

```
root@kali:~# mkfs /dev/md0p1
mke2fs 1.45.4 (23-Sep-2019)
Creating filesystem with 102400 1k blocks and 25688 inodes
Filesystem UUID: 0fb66c20-28fc-49f6-8727-8b88eadb949a
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

root@kali:~# mkdir /mnt/raid1
root@kali:~# mount /dev/md0p1 /mnt/raid1
```

Creamos un fichero de 50 MB

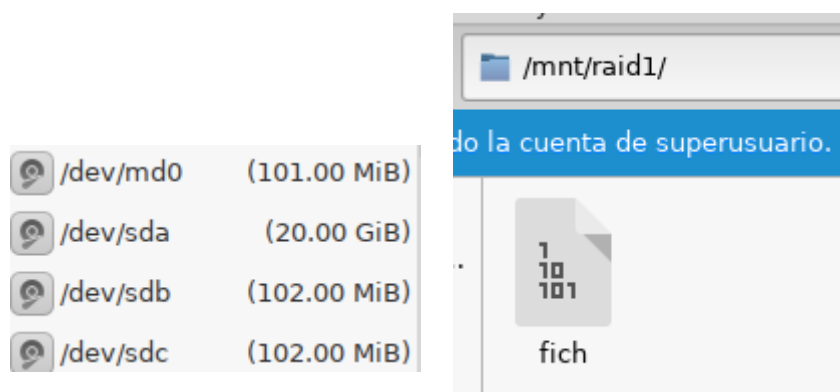
```
# dd if=/dev/zero of=/mnt/raid1/fich bs=512 count=100000
```

```
# ls -l /mnt/raid1
```

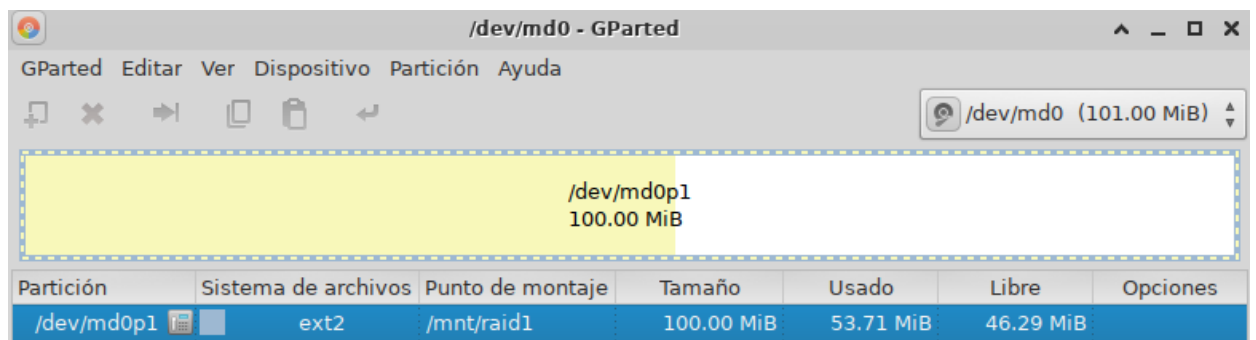
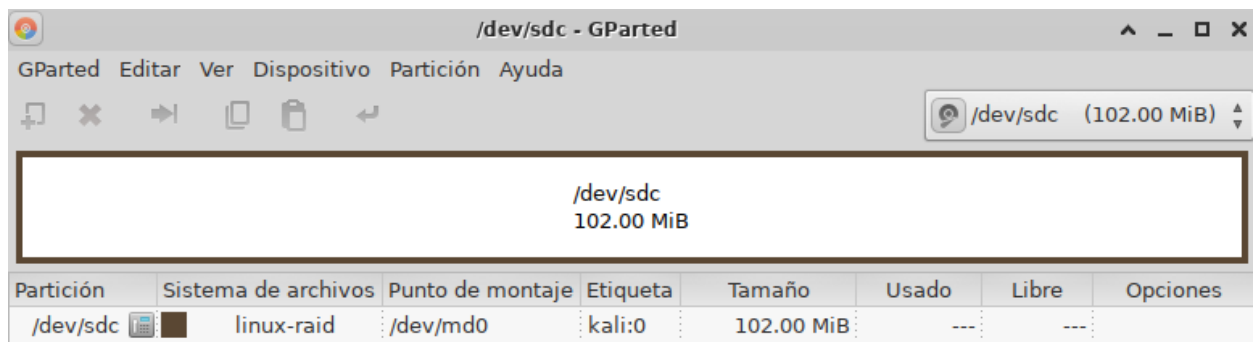
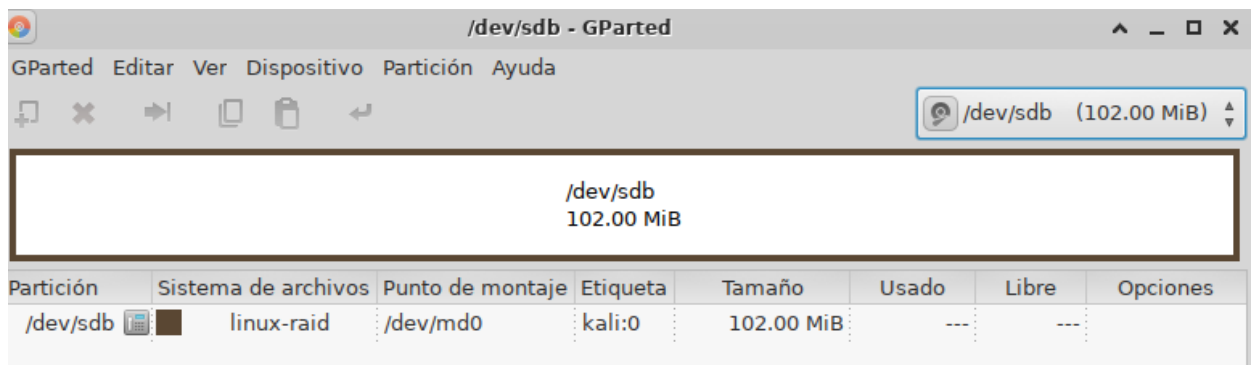
El **primer comando** crea el **fichero** de **50 MB** aproximadamente (10 000 bloques de 512 bytes). Con el segundo comprobamos el resultado.

```
root@kali:~# dd if=/dev/zero of=/mnt/raid1/fich bs=512 count=100000
100000+0 registros leídos
100000+0 registros escritos
51200000 bytes (51 MB, 49 MiB) copied, 0,47528 s, 108 MB/s
```

```
root@kali:~# ls -l /mnt/raid1
total 50209
-rw-r--r-- 1 root root 51200000 ene 12 14:29 fich
drwx----- 2 root root 12288 ene 12 14:22 lost+found
```



Con **Gparted**, comprobamos en un modo gráfico las particiones.



Para **comprobar** el **estado** del **RAID** tenemos el fichero `/proc/mdstat`. En nuestro caso nos muestra que tenemos un RAID md0, que está activo, de tipo raid1, cuyos componentes son sdc y sdb.

```
root@kali:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md0 : active raid1 sdc[1] sdb[0]
      103424 blocks super 1.2 [2/2] [UU]

unused devices: <none>
```

Si un disco **falla**, podemos **quitarlo** y el **RAID se mantiene** porque espera que lo sustituyamos por otro. Para ello primero hay que **marcarlo** como **disco fallido** y luego **quitarlo** del RAID. Vamos a **hacerlo** con el **disco sdb**. Los comandos son:

```
# mdadm /dev/md0 --fail /dev/sdb
```

```
# mdadm /dev/md0 --remove /dev/sdb
```


Después de cada comando consultamos el **mdstat** para comprobar qué ha ocurrido. En el primer caso se indica que el disco **sdb** ha fallado (**F de Failed**).

```
root@kali:~# mdadm /dev/md0 --fail /dev/sdb
mdadm: set /dev/sdb faulty in /dev/md0
```

```
root@kali:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md0 : active raid1 sdc[1] sdb[0](F)
      103424 blocks super 1.2 [2/1] [_U]
mensaje
unused devices: <none>
```

En el segundo comando ya no aparece sdb. En ambos casos la composición del RAID 1 aparece incompleta [_ U].

```
root@kali:~# mdadm /dev/md0 --remove /dev/sdb
mdadm: hot removed /dev/sdb from /dev/md0
root@kali:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md0 : active raid1 sdc[1]
      103424 blocks super 1.2 [2/1] [_U]
mensaje
unused devices: <none>
```

Sin embargo, los **datos siguen** ahí. Si quisiéramos que el disco **sdb volviera** al RAID, primero habría que **eliminar su configuración anterior** (borrar el superbloque) y **luego añadirlo**. Los comandos son:

```
# mdadm --zero-superblock /dev/sdb
```

```
# mdadm /dev/md0 --add /dev/sdb
```

Como se ve en la Figura, después de añadir el disco se inicia un proceso de sincronización (recovery). Hasta que no ha terminado el RAID no recupera el estado [UU].

Anteriormente sdb tenía un valor [0], y ahora pasa a tener un valor [2]

```
root@kali:~# mdadm --zero-superblock /dev/sdb
root@kali:~# mdadm /dev/md0 --add /dev/sdb
mdadm: added /dev/sdb
root@kali:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md0 : active raid1 sdb[2] sdc[1]
      103424 blocks super 1.2 [2/2] [UU]
mensaje
unused devices: <none>
```

Si queremos que los **sistemas de ficheros** creados en un RAID estén **disponibles al arrancar**, debemos **incluirlos** en el **fstab**, como es habitual.

Podemos **desactivar temporalmente** un RAID.

```
# mdadm --stop /dev/md0
```

El comando anterior necesita que el disco no esté siendo utilizado en ningún sistema de ficheros. Para ello lo tendremos que desmontar y ya podremos pararlo.

Activamos el RAID.

mdadm --assemble --scan

```
root@kali:~# mdadm --stop /dev/md0
mdadm: Cannot get exclusive access to /dev/md0:Perhaps a running process, mounted filesystem or active volume group?
root@kali:~# df
S.ficheros    bloques de 1K  Usados  Disponibles  Uso%  Montado en
udev          940004      0      940004      0%    /dev
tmpfs         194808      5940    188868      4%    /run
/dev/sda1     19478160  5889924  12575756   32%    /
tmpfs         974024      0      974024      0%    /dev/shm
tmpfs         5120        0       5120      0%    /run/lock
tmpfs         974024      0      974024      0%    /sys/fs/cgroup
tmpfs         194804      12     194792      1%    /run/user/0
/dev/md0p1    99150      51747    42283     56%    /mnt/raid1
root@kali:~# umount /dev/md0p1
root@kali:~# mdadm --stop /dev/md0
mdadm: stopped /dev/md0
root@kali:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
unused devices: <none>
root@kali:~# mdadm --assemble --scan
mdadm: /dev/md/0 has been started with 2 drives.
root@kali:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md0 : active (auto-read-only) raid1 sdb[2] sdc[1]
      103424 blocks super 1.2 [2/2] [UU]
unused devices: <none>
```

Pero como podemos comprobar está activo, pero no está montado. Hay que montarlo de nuevo.

```
root@kali:~# df
S.ficheros    bloques de 1K  Usados  Disponibles  Uso%  Montado en
udev          940004      0      940004      0%    /dev
tmpfs         194808      5932    188876      4%    /run
/dev/sda1     19478160  5889924  12575756   32%    /
tmpfs         974024      0      974024      0%    /dev/shm
tmpfs         5120        0       5120      0%    /run/lock
tmpfs         974024      0      974024      0%    /sys/fs/cgroup
tmpfs         194804      12     194792      1%    /run/user/0
root@kali:~# mount /dev/md0p1 /mnt/raid1
root@kali:~# df
S.ficheros    bloques de 1K  Usados  Disponibles  Uso%  Montado en
udev          940004      0      940004      0%    /dev
tmpfs         194808      5932    188876      4%    /run
/dev/sda1     19478160  5889924  12575756   32%    /
tmpfs         974024      0      974024      0%    /dev/shm
tmpfs         5120        0       5120      0%    /run/lock
tmpfs         974024      0      974024      0%    /sys/fs/cgroup
tmpfs         194804      12     194792      1%    /run/user/0
/dev/md0p1    99150      51747    42283     56%    /mnt/raid1
```