



PRACTICA 6 – RAID LINUX

1. Qué es RAID y que significa?

RAID es un acrónimo del inglés que significa Redundant Array of Independent Disks, literalmente «matriz de discos independientes redundantes». Debes saber que no todas estas configuraciones ofrecen redundancia, es depende del caso. Además, existen diferentes tipos o maneras en las que se ofrece redundancia.

La idea primaria de las configuraciones RAID es proteger la información almacenada en la unidad de almacenamiento ante fallos. Aunque hay diferentes configuraciones que obvian esta parte y solo buscan mejorar las velocidades de transferencia. Incluso, existen configuraciones avanzadas que ofrecen ambas características.

Fundamentalmente, lo que se hace es generar un único volumen de unidades de almacenamiento que funcionan en conjunto. Básicamente, tenemos dos tipos de configuraciones, como son las siguientes:

- **Disk mirroring:** Es un tipo de configuración RAID que busca redundancia de datos ante un posible fallo en una de las unidades de almacenamiento
- **Disk stripping:** Esta configuración RAID que no busca redundancia, sino conseguir mayores velocidades de transferencia de datos

2. Cómo funciona un RAID de discos duros

Un sistema RAID funciona emplazando los datos en varios discos duros, y permitiendo que las operaciones de entrada y salida (I/O) funcionen de manera balanceada, mejorando el rendimiento. En otras palabras, o bien los datos se escriben en ambos discos al mismo tiempo, o bien se escribe un dato en uno, y otro dato en otro para repartir el trabajo. Los sistemas RAID se presentan en el sistema operativo como si fueran un único disco lógico, dado que consisten en un solo volumen.

Dicho de otra manera, lo que hacen es almacenar una información dividida entre varias unidades distintas. De esta manera se accede a ellos en paralelo y se duplica la velocidad de acceso. Sin embargo, en el caso de los discos duros esta función no se suele encontrar en los PC domésticos.

Para que un sistema RAID funcione es necesaria la presencia de una controladora RAID, y puede ser o bien por hardware, o bien por software. A día de hoy, la gran mayoría de PC de usuario ya cuentan con una controladora RAID por software integrada en la BIOS de la placa base, y de hecho las **controladoras por hardware** tan solo se usan en entornos empresariales a día de hoy.



3. Configuraciones RAID de tipo simple

Actualmente existen muchos tipos de configuraciones de este tipo. Las más típicas y soportadas por cualquier controladora de placa base del mercado son las RAID 0, 1 y 5. Pero, existen muchos otros tipos de configuraciones que vienen a «solucionar» diferentes problemas que nos pueden surgir en el día a día cuando mantenemos y manipulamos un sistema de estas características.

Dentro de este primer apartado veremos las configuraciones que se denominan simples. Estas son diferentes tipos de configuraciones basadas en configuración única y más adelante veremos las configuraciones anidadas, que son la combinación de varias de estas configuraciones simples.

RAID 0

Es un tipo de **configuración stripping**, por lo que no ofrece redundancia de datos. Para este tipo de configuración se requieren **al menos dos unidades** de almacenamiento y el número máximo viene determinado por la controladora. **Windows** y cualquiera sistema operativo **vera** esta configuración cómo si fuera un **único disco duro**, sin importar de la cantidad de unidades instaladas.

Lo que se hace en esta configuración es que los **datos se escriben secuencialmente**. Si tenemos dos discos, uno denominado HDD1 y el otro HDD2, lo que hace es partir la información y escribirla primero en el HDD1 y luego en el HDD2, y así hasta almacenar toda la información.

Gracias a este sistema se permite **prácticamente duplicar la velocidad** de escritura y también, de lectura. Debemos tener en cuenta que no se suman las velocidades. Si cada unidad ofrece una velocidad de escritura de 2500 MB/s, en esta configuración teóricamente deberían ofrecer 5000 MB/s de escritura. La realidad es que se **pierde entre el 10-25% de rendimiento**.

Pero, esta configuración tiene un grave problema. **Si cualquiera de las unidades usadas en la RAID 0 sufre un daño, toda la información se perderá** sin remedio por lo que, tal vez, prefieras decantarte por otro estándar más fiable en ese sentido.

RAID 1

Pasamos a la primera **configuración de tipo mirroring o de espejo**. Este tipo de configuración busca realizar una **copia de seguridad de los datos** almacenados. La información se escribe de manera simultáneamente en los discos duros. Se requiere de **al menos dos unidades** y el límite está determinado por la controladora por o que ganaos, respecto del Raid 0, ese punto de tener a salvo la información mientras una de las unidades esté operativa y funcional.

Dicha configuración tiene un gran problema. Aunque lo normal es pensar que primero la **información se guarda en el disco duro primario y luego se copia en el secundario**, esto no es así. La realidad es que la información se escribe en los dos discos duros al mismo tiempo. Esto supone una **reducción en la velocidad de escritura de hasta un 25%**, pero la velocidad de lectura podría subir en hasta un 25%.



Lo bueno de esta configuración es que, si se **daña una de las unidades**, se puede **recuperar la información** almacenada. No es habitual, pero ambas unidades se podrían dañar al mismo tiempo, en ese caso, la información se pierde y no tendremos opción de volver a recuperarla de ninguna manera. Eso sí, estamos hablando de una de las casuísticas menos habituales de las que podemos sufrir al utilizar uno de estos sistemas.

Algo que debemos tener en cuenta es que las dos unidades de almacenamiento deben tener la misma capacidad. Si emparejamos un disco duro de 500 GB con uno de 1 TB, solo se utilizarán 500 GB de la unidad de 1 TB, independientemente de si es primaria o secundaria, por lo que estaríamos desperdiciando un espacio que podría utilizarse para otra cosa. Así que recordad instalar siempre componentes con la misma capacidad.

RAID 5

Esta configuración de discos duros con **paridad distribuida es bastante utilizada en la actualidad**. Las unidades de **almacenamiento se dividen en bloques** donde se almacena la información. Dentro del conjunto de bloques, uno se destina a paridad, para garantizar la redundancia de datos.

La paridad lo que nos permite es **reconstruir los datos guardados** en un disco duro, si este termina fallando. El bloque de paridad, por otro lado, se almacena en un disco duro distinto para poder restaurar la información.

Se requiere para este tipo de configuración de **al menos tres discos duros** para así poder garantizar la redundancia de los datos.

Tiene un gran inconveniente este sistema y es que **solo permite el fallo de un disco duro a la vez**. Si por cualquier casualidad se dañaran dos de las unidades de almacenamiento que conforman la RAID 5, la información se perderá. Además, si se diera un error durante el proceso de restauración, toda la información almacenada se perderá.

RAID LINUX

El software RAID incluido en las versiones actuales de Linux (y Ubuntu) se basa en el comando «**mdadm**» y funciona muy bien (mejor incluso que los llamados controladores RAID «hardware»)

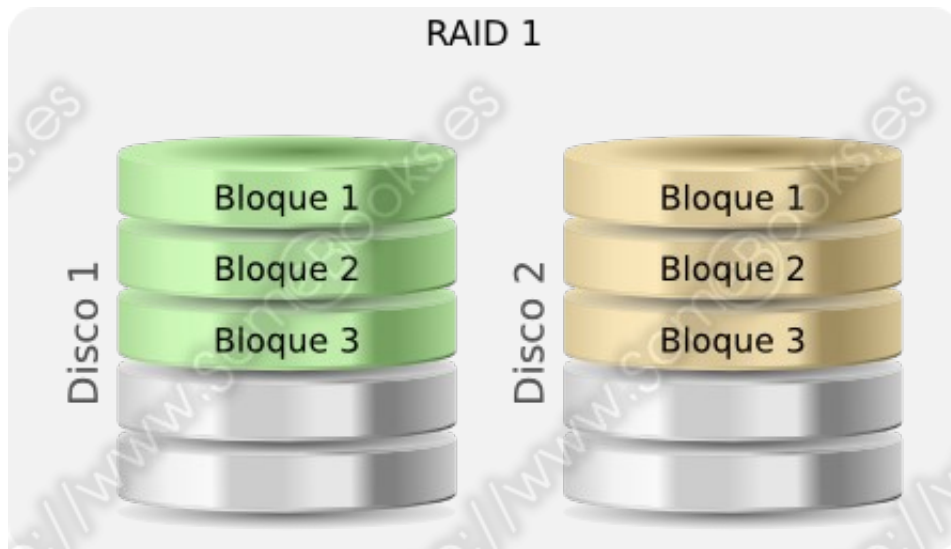
mdadm es la herramienta que utilizaremos para administrar los sistemas RAID en Linux (lo haremos todo como root). Si no está instalada, la tendremos que instalar :

apt install mdadm

- consulta el manual de dicho comando
- **/etc/mdadm/mdadm.conf** es el archivo de configuración de mdadm,
- Podemos visualizar el estado del RAID con el siguiente comando: **mdadm --detail --scan** o también con **cat /proc/mdstat** o **watch -n 1 cat /proc/mdstat**

Una vez que creamos un RAID, podemos también guardar la información en el archivo de configuración con: **mdadm --detail --scan >**

/etc/mdadm/mdadm.conf



SECUENCIA/DESARROLLO:

1. Crear un RAID 1 bajo Linux

PASO 1: Instalar dos discos duros de capacidad 10 GB

Primer disco – sdb

Segundo disco – sdc

PASO 2: Visualización del archivo /proc/mdstat para ver estado del RAID

cat /proc/mdstat

Para hacerlo constantemente: watch -n1 cat /proc/mdstat (Ctrl+c para parar la orden watch)

```
usadmin04@ubsrv04:~$ cat /proc/mdstat
Personalities : [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
unused devices: <none>
usadmin04@ubsrv04:~$ _
```

PASO 3: Preparación del disco. Particionamiento.

Crear una partición primaria de tipo **Linux raid autodetect** (identificador 29) en /dev/sdb, /dev/sdc asignándole todo el espacio disponible (previamente se debe crear la tabla de particiones)

fdisk /dev/sdb (también valdría con cfdisk o parted)

- Command: n (Nuevo)



- Command: p (Primario)
- Command: 1 (Nº de particiones)
- Command: t (Cambiar de tipo)
- Poner tipo linux raid auto (fd)
- Command: p (Tabla de partición).
- Command: w (Guardar)
- fdisk /dev/sdc (Realizar los mismos pasos).
- Podemos comprobar los nuevos dispositivos disponibles con lsblk

```
usadmin04@ubsrv04:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0   60G  0 disk
├─sda1       8:1    0    1M  0 part
├─sda2       8:2    0    1G  0 part /boot
├─sda3       8:3    0    4G  0 part [SWAP]
├─sda4       8:4    0    2G  0 part /tmp
├─sda5       8:5    0   10G  0 part /var
├─sda6       8:6    0    3G  0 part /home
└─sda7       8:7    0   40G  0 part /
sdb          8:16   0   10G  0 disk
└─sdb1       8:17   0   10G  0 part
sdc          8:32   0   10G  0 disk
└─sdc1       8:33   0   10G  0 part
sr0         11:0    1 1024M  0 rom
usadmin04@ubsrv04:~$ _
```

PASO 4: Creación de un raid 1.

***mdadm --create /dev/md/md_raid1 --level=1 - -raid-devices=2 /dev/sdb1
/dev/sdc1***

(podemos añadir el el parámetro `--verbose` para que nos vaya mostrando información)

Verificar el estado de /proc/mdstat (opcionalmente con mdadm --detail)

PASO 5: Formatearlo como ext4

mkfs.ext4 /dev/md/md_raid1

Verificar el estado de /proc/mdstat (opcionalmente con mdadm --detail)

PASO 6: Guardar detalles del RAID en mdadm.conf

mdadm --detail --scan >> /etc/mdadm/mdadm.conf"

Visualizar este archivo



```

root@ubsrv04:~# cat /etc/mdadm/mdadm.conf
# mdadm.conf
#
# !NB! Run update-initramfs -u after updating this file.
# !NB! This will ensure that initramfs has an uptodate copy.
#
# Please refer to mdadm.conf(5) for information about this file.
#
# by default (built-in), scan all partitions (/proc/partitions) and all
# containers for MD superblocks. alternatively, specify devices to scan, using
# wildcards if desired.
#DEVICE partitions containers

# automatically tag new arrays as belonging to the local system
HOMEHOST <system>

# instruct the monitoring daemon where to send mail alerts
MAILADDR root

# definitions of existing MD arrays

# This configuration was auto-generated on Tue, 05 Aug 2025 17:02:17 +0000 by mkconf
ARRAY /dev/md/md_raid1 metadata=1.2 UUID=fe6e9887:ed574c78:11105f2e:57cf456d
root@ubsrv04:~# _

```

PASO 7: Montar el RAID

- Primero debes crear el directorio /media/datos

mount /dev/md/md_raid1 /media/datos

Comprobamos espacio → **df -Th /media/datos**

```

root@ubsrv04:~# df -Th /media/datos
Filesystem      Type  Size  Used Avail Use% Mounted on
/dev/md127      ext4  9,8G   24K  9,3G   1% /media/datos
root@ubsrv04:~#

```

Captura pantalla de comprobación del estado del RAID

```

root@ubsrv04:~# cat /proc/mdstat
Personalities : [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md127 : active raid1 sdc1[1] sdb1[0]
        10474496 blocks super 1.2 [2/2] [UU]

unused devices: <none>
root@ubsrv04:~#

```

PASO 8: Simular fallos en el RAID y arreglarlos. Para ello, tenemos que realizar :

- Desconexión de un disco mdadm, vigilando lo que pasa en /proc/mdstat y volver a instalar el disco y añadir en el RAID



`mdadm -- manage -- set-faulty /dev/md/md_raid1 /dev/sdb1 (--set-faulty , --fail)`

- Podemos observar el fallo:
 - `watch -n1 cat /proc/mdstat`
 - `mdadm -- detail /dev/md127 // /dev/md/md_raid1`

```
root@ubsrv04:~# cat /proc/mdstat
Personalities : [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md127 : active raid1 sdc1[1] sdb1[0](F)
      10474496 blocks super 1.2 [2/1] [_U]

unused devices: <none>
root@ubsrv04:~#
```

```
root@ubsrv04:~# mdadm --detail /dev/md127
/dev/md127:
    Version : 1.2
  Creation Time : Tue Sep 23 07:30:34 2025
    Raid Level : raid1
    Array Size : 10474496 (9.99 GiB 10.73 GB)
  Used Dev Size : 10474496 (9.99 GiB 10.73 GB)
    Raid Devices : 2
  Total Devices : 2
 Persistence : Superblock is persistent

    Update Time : Tue Sep 23 07:53:18 2025
      State : clean, degraded
Active Devices : 1
Working Devices : 1
Failed Devices : 1
Spare Devices : 0

Consistency Policy : resync

        Name : ubsrv04:md_raid1 (local to host ubsrv04)
        UUID : fe6e9887:ed574c78:11105f2e:57cf456d
        Events : 19

   Number   Major   Minor   RaidDevice State
    -       -       -       -        -
     0       0       0         0        removed
     1       8       33         1        active sync  /dev/sdc1
     0       8       17         -        faulty   /dev/sdb1
root@ubsrv04:~# _
```

- Reconstruir el RAID:
- Sacar el disco erróneo del RAID

`mdadm /dev/md127 --remove /dev/sdb1 (-r , --remove)`
- En este punto limpiamos la información de /dev/sdb1 (ponemos a cero el superbloque) para “resetear” el disco; para luego añadirlo como un disco



nuevo-vacio.

`mdadm --zero-superblock /dev/sdb1`

- Añadir de nuevo en el RAID

`mdadm /dev/md0 -av /dev/sdb1`

(`-a / --add`) `-v --verbose`)

- **Nota:** si intentas limpiar el disco `/dev/sdb1` (`--zero-superblock`) sin sacarlo te dará mensaje que el disco está ocupado). Entonces es cuando tienes que parar el Raid:

`mdadm --stop /dev/md/md_raid1`

- Y luego volverlo a ensamblar:

`mdadm --assemble /dev/md/md_raid1`

(asi funciona sólo si se guardó previamente la configuración del RAID en el fichero de configuración)

```
root@ubsrv04:~# mdadm --zero-superblock /dev/sdb1
root@ubsrv04:~# mdadm /dev/md0 -av /dev/sdb1
mdadm: error opening /dev/md0: No such file or directory
root@ubsrv04:~# mdadm /dev/md/md_raid1 -av /dev/sdb1
mdadm: added /dev/sdb1
root@ubsrv04:~# mdadm --stop /dev/md/md_raid1
mdadm: Cannot get exclusive access to /dev/md/md_raid1:Per
root@ubsrv04:~# mdam --assemble /dev/md/md_raid1
Command 'mdam' not found, did you mean:
  command 'mda' from deb mailutils-mda (1:3.16-1build1)
  command 'mdadm' from deb mdadm (4.3-1ubuntu2.1)
Try: apt install <deb name>
root@ubsrv04:~# mdadm --assemble /dev/md/md_raid1
root@ubsrv04:~# _
```

PASO 9: Configurar para que el RAID se monte automáticamente al iniciar el equipo

Reinicia el sistemas → Verás que no está montado el RAID

Añade la línea a `fstab` para que se monte automáticamente al inicio

DOCUMENTACIÓN

Para montar un RAID en el archivo `/etc/fstab`, necesitas añadir una línea que especifique el dispositivo RAID y su punto de montaje, junto con el tipo de sistema de archivos y las opciones de montaje.

`/dev/md0 /mnt/raid1 ext4 defaults 0 0`



- `/dev/md0`: Es el dispositivo RAID que has creado (ajusta esto según el nombre de tu dispositivo RAID).
- `/mnt/raid1`: Es el punto de montaje donde deseas que el RAID esté disponible. Asegúrate de que el directorio exista; si no, puedes crearlo con `mkdir`.
- `ext4`: Es el tipo de sistema de archivos. Cambia esto si utilizas otro sistema de archivos (por ejemplo, `xfs`, `btrfs`, etc.).
- `defaults`: Son las opciones de montaje. Puedes personalizarlas según tus necesidades, pero `defaults` es una opción común que incluye las configuraciones estándar.
 - `0`: Indica si el sistema de archivos debe ser respaldado por `dump` (0 significa que no).
 - `0`: Indica el orden en que `fsck` debe verificar los sistemas de archivos al inicio (0 significa que no se debe verificar).

Por último debes montar todos los sistemas de archivos definidos en `/etc/fstab` :

Captura pantallas de comprobación

`sudo mount -a`

```
root@ubsrv04:~# sudo mount -a
[ 3514.026328] /dev/md0: Can't lookup blockdev
mount: /mnt/raid1: special device /dev/md0 does not exist.
dmesg(1) may have more information after failed mount system call.
mount: (hint) your fstab has been modified, but systemd still uses
the old version; use 'systemctl daemon-reload' to reload.
root@ubsrv04:~# _
```

Compruébalo reiniciando el sistema + `lsblk -f`

```
usadmin04@ubsrv04:~$ lsblk -f
```

NAME	FSTYPE	FSVER	LABEL	UUID	FSAVAIL	FSUSE%	MOUNTPPOINTS
sda							
└sda1							
└sda2	ext4	1.0		87b09cf4-e270-47b3-bd2d-b36030af24a6	805,9M	10%	/boot
└sda3	swap	1		990f4832-6cea-462f-911e-3ff5ff8c5794			[SWAP]
└sda4	ext4	1.0		2a306b43-f751-47ef-8038-70930f2ffef4	1,8G	0%	/tmp
└sda5	ext4	1.0		533797b0-a1ba-4290-bd90-d2745b06e8ce	8,8G	5%	/var
└sda6	ext4	1.0		134a7ef8-46e0-4144-a555-df7f7010894a	2,7G	0%	/home
└sda7	ext4	1.0		45748dbb-dd27-4bcd-a626-065c9632343b	34,8G	6%	/
sdb							
└sdb1	linux_raid_member	1.2	ubsrv04:md_raid1	fe6e9887-ed57-4c78-1110-5f2e57cf456d			
└md127	ext4	1.0		e7ba1c61-562c-4165-ae59-1c8577759659			
sdc							
└sdc1	linux_raid_member	1.2	ubsrv04:md_raid1	fe6e9887-ed57-4c78-1110-5f2e57cf456d			
└md127	ext4	1.0		e7ba1c61-562c-4165-ae59-1c8577759659			
sr0							

```
usadmin04@ubsrv04:~$
```



Obtenemos los detalles del RAID : mdadm --query /dev/md/md_raid1

```
usadmin04@ubsrv04:~$ sudo mdadm --query /dev/md/ubsrv04\:md_raid1
/dev/md/ubsrv04:md_raid1: 9.99GiB raid1 2 devices, 0 spares. Use mdadm --detail for more detail.
usadmin04@ubsrv04:~$
```

2. Opcional: Crear RAID 5. (Para ello debes añadir los discos que necesites)