

Construyendo un RAID 10 en linux

Para esta práctica vamos a tener una máquina virtual con 5 discos, 1 de sistema y otros 4 para usar en la configuración **RAID 10**, cada uno con 8gb, a efecto de demostración.

En este caso, el sistema operativo estará en */dev/sda* y sus particiones, mientras que los discos para los datos estarán en */dev/sdb*, */dev/sdc*, */dev/sdd*, */dev/sde*.

```
root@server:~# ls /dev/sd* -l
/dev/sda
/dev/sda1
/dev/sdb
/dev/sdc
/dev/sdd
/dev/sde
root@server:~#
```

Creación del dispositivo RAID 10

Empezamos instalando el controlador de **RAID** por software: mdadm, como se ha realizado en anterior práctica.

Con las herramientas instaladas, procedemos a crear un */dev/md0* que será nuestro **disco RAID**, indicando el nivel **RAID 10** y los 4 discos reales que van a formarlo.

```
root@server:~# mdadm -v --create /dev/md0 --level=raid10 --raid-devices=4 /dev/sdb
/dev/sdc /dev/sdd /dev/sde
mdadm: layout defaults to n2
mdadm: layout defaults to n2
mdadm: chunk size defaults to 512K
mdadm: size set to 8380416K
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
root@server:~#
```

Para que ese array de discos sea reconocido en cada inicio del sistema, hay que añadir en */etc/mdadm/mdadm.conf* la información relacionada al array, de la misma forma que la tengamos en este momento.

```
root@server:~# mdadm --detail --scan --verbose >> /etc/mdadm/mdadm.conf
root@server:~#
```

Y ya tenemos nuestro dispositivo **RAID 10**.

Preparación del dispositivo

Ahora disponemos de un **RAID 10** de 4 discos de 8gb, que corresponden a una capacidad total de 16gb utilizables, como el dispositivo `/dev/md0`.

Este dispositivo es transparente para nosotros y no es diferente de cualquier otro dispositivo de bloques, con lo que se puede particionar, formatear, etc.

Para esta demostración, se creará una única partición que ocupe todo el disco y que será montada en `/data`.

Así pues, sin mas preámbulo la particionamos con `fdisk`. Este es el resultado:

```
root@server:~# fdisk -l /dev/md0
```

```
Disco /dev/md0: 16 GiB, 17163091968 bytes, 33521664 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 524288 bytes / 1048576 bytes
Tipo de etiqueta de disco: gpt
Identificador del disco: E3FE7B0A-0F5D-4151-84E8-49670C33B65E
```

```
Device   Start   End   Sectors Size Type
/dev/md0p1 2048 33521630 33519583 16G Linux filesystem
```

```
root@server:~#
```

La primera (y única partición) se llama `/dev/md0p1` y es el dispositivo que vamos a formatear, para posteriormente montarlo.

```
root@server:~# mkfs.ext4 /dev/md0p1
```

```
mke2fs 1.42.12 (29-Aug-2014)
```

Se está creando El sistema de ficheros con 4189947 4k bloques y 1048576 nodos-i

UUID del sistema de ficheros: 11e454ce-72c4-41f8-a7bc-4d4a78b873c0

Respaldo del superbloque guardado en los bloques:

32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
4096000

Reservando las tablas de grupo: hecho

Escribiendo las tablas de nodos-i: hecho

Creando el fichero de transacciones (32768 bloques): hecho

Escribiendo superbloques y la información contable del sistema de ficheros: hecho

```
root@server:~#
```

Creamos la carpeta que va a servir de *mountpoint* para esta nueva partición:

```
root@server:~# mkdir /data
root@server:~#
```

Añadimos la partición en el fichero */etc/fstab*, para que se monte automáticamente tras cada reinicio:

```
root@server:~# grep md0p1 /etc/fstab
/dev/md0p1 /data ext4 defaults 0 0
root@server:~#
```

Finalmente la montamos. Como esta información ya está en el fichero */etc/fstab* no es necesario especificar los detalles.

```
root@server:~# mount /data
root@server:~# df -h
S.ficheros  Tamaño Usados  Disp Uso% Montado en
/dev/sda1    2,0G  651M  1,2G  35% /
udev         10M    0  10M   0% /dev
tmpfs        50M   4,4M   46M   9% /run
tmpfs        124M    0  124M   0% /dev/shm
tmpfs        5,0M    0   5,0M   0% /run/lock
tmpfs        124M    0  124M   0% /sys/fs/cgroup
/dev/md0p1   16G   44M  15G   1% /data
root@server:~#
```

Como detalle, al no tratarse de una partición raíz de sistema operativo, no hace falta reservar bloques de emergencia; se trata de un 5% de la capacidad que podemos liberar (5% de 16gb son 800mb que podemos usar).

```
root@server:~# tune2fs -m 0 /dev/md0p1
```

tune2fs 1.42.12 (29-Aug-2014)

Se pone el porcentaje de bloques reservados a 0% (0 bloques)

```
root@server:~# df -h
```

S.ficheros	Tamaño	Usados	Disp	Uso%	Montado en
/dev/sda1	2,0G	651M	1,2G	35%	/
udev	10M	0	10M	0%	/dev
tmpfs	50M	4,4M	46M	9%	/run
tmpfs	124M	0	124M	0%	/dev/shm
tmpfs	5,0M	0	5,0M	0%	/run/lock
tmpfs	124M	0	124M	0%	/sys/fs/cgroup
/dev/md0p1	16G	44M	16G	1%	/data

```
root@server:~#
```

Verificación

Podemos ver la información de estado del array de discos con el mismo comando *mdadm*, como sigue:

```
root@server:~# mdadm --detail /dev/md0
```

```
/dev/md0:
```

```
Version : 1.2
```

```
Creation Time : Sat Dec 12 21:19:42 2015
```

```
Raid Level : raid10
```

```
Array Size : 16760832 (15.98 GiB 17.16 GB)
```

```
Used Dev Size : 8380416 (7.99 GiB 8.58 GB)
```

```
Raid Devices : 4
```

```
Total Devices : 4
```

```
Persistence : Superblock is persistent
```

```
Update Time : Sat Dec 12 21:30:11 2015
```

```
State : clean
```

```
Active Devices : 4
```

```
Working Devices : 4
```

```
Failed Devices : 0
```

```
Spare Devices : 0
```

```
Layout : near=2
```

```
Chunk Size : 512K
```

```
Name : server:0 (local to host server)
```

```
UUID : 217558a7:bc1cb1d4:9530ecda:ea477a6b
```

```
Events : 19
```

Number	Major	Minor	RaidDevice	State
0	8	16	0	active sync set-A /dev/sdb

1	8	32	1	active sync set-B	/dev/sdc
2	8	48	2	active sync set-A	/dev/sdd
3	8	64	3	active sync set-B	/dev/sde

root@server:~#

RESUMEN: Ahora tengo un disco doble de rápido, doble de capacidad y con doble copia de datos.