mdadm Linux software RAID

CONCEPTS

RAID is not a backup: it survives a disk failure, not deletion, corruption or a dead controller.

Disks / partitions components
↓ `--create`
/dev/md0 md array
↓ `mkfs`
Filesystem ext4, xfs, LVM...
↓ `mount`
In use watch /proc/mdstat

Create, grow and add start a background resync; the array is usable meanwhile but slower and not yet redundant.

Most commands need root and many write superblocks: double-check device names with lsblk first.

INSPECT

`cat /proc/mdstat`
Show arrays, members and sync progress

`mdadm --detail` *`<md>`*
Array state, level, members, UUID

`mdadm --examine` *`<dev>`*
Read the RAID superblock of a member

`mdadm --query` *`<dev>`*
Is this device an array or a member?

`mdadm --detail --scan`
One-line summary per active array

`mdadm --examine --scan`
Find arrays on member superblocks

`watch cat /proc/mdstat`
Follow sync progress live

CREATE

`mdadm --create` *`<md>`* **`--level=1 --raid-devices=2`** *`<dev1> <dev2>`*
Mirror (RAID1) of two devices

`mdadm --create` *`<md>`* **`--level=5 --raid-devices=3`** *`<dev1> <dev2> <dev3>`*
RAID5: striping with parity

CREATE (cont.)

`mdadm --create` *`<md>`* **`--level=6 --raid-devices=4`** *`<devs>`*
RAID6: survives two failures

`mdadm --create` *`<md>`* **`--level=0 --raid-devices=2`** *`<dev1> <dev2>`*
RAID0: striping, no redundancy

`mdadm --create` *`<md>`* **`--level=10 --raid-devices=4`** *`<devs>`*
RAID10: mirrored stripes

`mdadm --create ... --spare-devices=1` *`<devs>`*
Include a hot spare

`mdadm --create ... --bitmap=internal`
Write-intent bitmap, faster resync

Prefer partitions of type Linux RAID, or whole disks, of equal size; the smallest member sets the size.

FAIL, REPLACE, REPAIR

`mdadm` *`<md>`* **`--fail`** *`<dev>`*
Mark a member as faulty

`mdadm` *`<md>`* **`--remove`** *`<dev>`*
Remove a failed member

`mdadm` *`<md>`* **`--add`** *`<dev>`*
Add a disk; rebuild starts

`mdadm` *`<md>`* **`--re-add`** *`<dev>`*
Re-add a recently removed member

`mdadm` *`<md>`* **`--replace`** *`<dev>`* **`--with`** *`<spare>`*
Swap a disk while it is still healthy

`mdadm` *`<md>`* **`--fail detached --remove detached`**
Drop members whose device vanished

`mdadm --zero-superblock` *`<dev>`*
Erase RAID metadata from a disk

`echo check > /sys/block/` *`<md>`* **`/md/sync_action`**
Scrub the array for mismatches

After a disk swap, copy the partition layout first (sfdisk -d | sfdisk), then --add the partition.

ASSEMBLE AND STOP

`mdadm --assemble --scan`
Assemble arrays from config or superblocks

`mdadm --assemble` *`<md> <dev1> <dev2>`*
Assemble one array from given members

`mdadm --stop` *`<md>`*
Deactivate an array (unmount first)

`mdadm --assemble --run` *`<md> <devs>`*
Start even with missing members

GROW AND RESHAPE

`mdadm --grow` *`<md>`* **`--raid-devices=`***`<n>`*
Change the number of active members

`mdadm --grow` *`<md>`* **`--size=max`**
Use all space after bigger members

After growing, extend the filesystem (resize2fs, xfs_growfs) or the LVM layer on top.

CONFIG AND MONITORING

`mdadm --detail --scan >> /etc/mdadm/mdadm.conf`
Persist array definitions

`mdadm --monitor --scan --daemonise`
Run the monitor in the background

`systemctl status mdmonitor`
Check the monitor service

`update-initramfs -u`
Refresh initramfs (Debian/Ubuntu)

RECIPES

Mirror two disks and mount

```
sudo mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/sdb1 /dev/sdc1
sudo mkfs.ext4 /dev/md0
sudo mount /dev/md0 /mnt
sudo mdadm --detail --scan | sudo tee -a /etc/mdadm/mdadm.conf
```