



RAID met de Sheevaplug

Lieven Baes

15 april 2010

Wat is raid?

RAID = Redundant **A**rray of **I**nexpensive **D**isks

RAID = Redundant Array of **I**ndependent Disks



Data opslaan op **meerdere** harde schijven met als doel:

- ▶ veel, goedkope en veilige opslagruimte creëren,
- ▶ snelheid van de data-overdracht verhogen,
- ▶ redundant data opslaan (dezelfde data op meerdere schijven),
- ▶ de beschikbaarheid van de data verhogen (een vorm van backup),
- ▶ bij uitval, een andere schijf erin, syncen en huppekee,
- ▶ OS om zeep, live-systeem erin, data bereikbaar . . .

1 Hardware raid controllers:

Een elektronische chip zorgt ervoor dat de meerdere harde schijven gezien worden als 1 device (harddisk).



▶ Voordeel:

- ▶ Snel datadoorstroom,
- ▶ De chip doet de berekeningen en ontlast het OS,
- ▶ Duurdere controllers zijn hot-swappable.

▶ Nadeel:

- ▶ Wat als de controller sneuvelt???
- Zoek maar een identieke of compatibele controller!
- ▶ Wordt de driver ondersteund?
- ▶ Duur, er bestaan ook goedkopere controllers.

2 Software raid:

Het besturingssysteem (OS) verzorgt het rekenwerk.

(Linux, Windows, Solaris, BSD's, ...)

- ▶ **Voordeel:**

- ▶ goedkoop,
- ▶ volledig hardware onafhankelijk,
- ▶ op alle “gelijkaardige” OS'en bereikbaar.

- ▶ **Nadeel:**

- ▶ minder snel, afh. van de CPU en de belasting van het OS,
- ▶ het OS moet wat meer werken,
- ▶ OS-afhankelijk.

Met de hedendaagse kracht van de CPU's heeft software raid veelal de voorkeur.

```
janu@kloon:~$ cat /proc/mdstat
Personalities : [raid0] [raid1] [raid5] [raid6] [raid5] [raid4]
md4 : active raid5 sdd1[0] sde1[2] sdd1[1]
      976767872 blocks level 5, 64k chunk, algorithm 2 [3/3] [UU]

md7 : active raid1 sda5[0] sdb5[1]
      29991616 blocks [2/2] [UU]

md6 : active raid1 sda3[0] sdb3[1]
      85939648 blocks [2/2] [UU]

md5 : active raid1 sda2[0] sdb2[1]
      40041920 blocks [2/2] [UU]

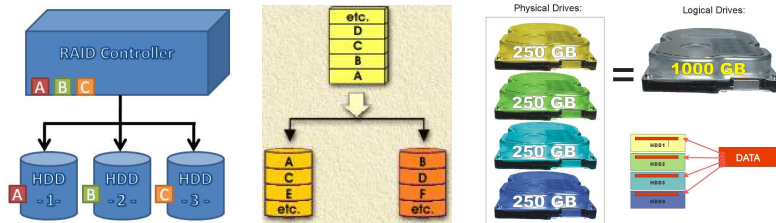
md1 : active raid0 sda1[0] sdb1[1]
      995712 blocks 64k chunks

unused devices: (none)
janu@kloon:~$
```



Soorten RAID vormen

1 RAID 0 of striping:

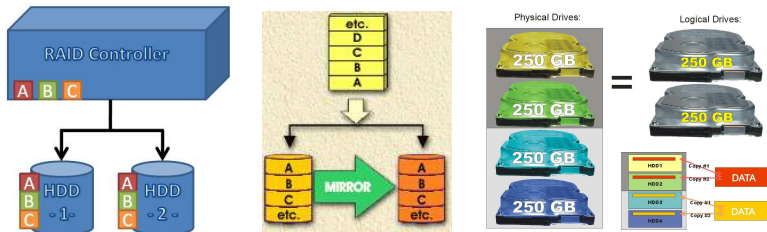


De data wordt in stukken over de verschillende schijven verdeeld. Er kan terzelfdertijd naar meerdere schijven geschreven worden.

Gevolgen:

- ▶ Zeer snelle datastroom (dubbel, bij 2 schijven, ...),
- ▶ Geen foutcorrecties; als 1 disk uitvalt, is alle data verloren,
- ▶ Geen ruimteverlies:
Totale capaciteit = de som van de afzonderlijk capaciteiten van de HD's.

2 RAID 1 of mirroring:

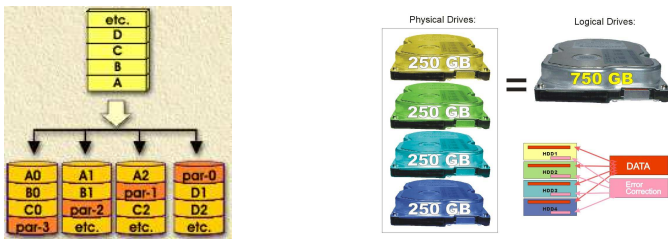


Dezelfde data wordt gelijktijdig naar de 2 disken geschreven.
De 2 disken zijn een copy van elkaar.

Gevolgen:

- ▶ De helft van de fysieke capaciteit is beschikbaar,
- ▶ Volledige redundantie; als 1 disk uitvalt, dan blijft alle data behouden:
Bij uitval van 1 disk, werkt het systeem gewoon verder.
- ▶ Veilig + "directe" backup.

3 RAID 5 (striping + parity-check):



De data wordt verdeeld over de schijven, samen met een parity-check.

De parity-checks worden verdeeld over de verschillende schijven.

Gevolgen:

- ▶ De capaciteit verhoogd als het aantal fysieke HD's stijgt ($n - 1$),
- ▶ Volledige redundantie; als 1 disk uitvalt, kan alle data terug opgebouwd worden (syncing),
- ▶ Schrijf-opdrachten duren langer dan lees-opdrachten (parity-check).

4 Andere, minder gebruikte raid-systemen:

RAID 2, RAID 3, RAID 4, RAID 6, Linear RAID

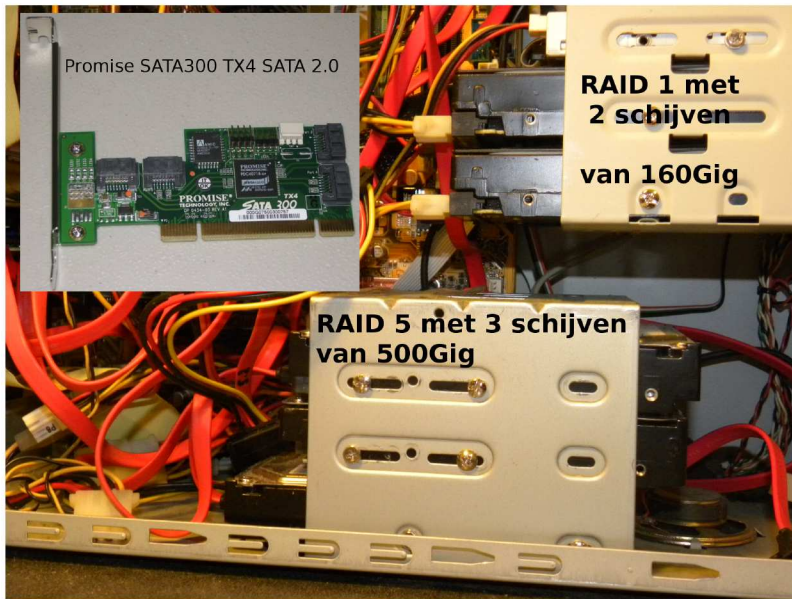
Combinaties:

RAID 1+0, RAID 5+0

Meer informatie, zie:

<http://www.fujitsu.com/global/services/computing/storage/eternus/glossary/raid/>

Voorbeeld van een RAID-configuratie met



Software RAID op een linux-pc : **raid-kernel-modules**

De nodige **raid-modules** zijn meestal reeds aanwezig in de kernel:

```
yanu@kloon:~/eigen/linuxbabbel$ lsmod | grep raid
raid456                42408  1
raid1                  16099  3
raid0                  5585   1
md_mod                 67165  8 raid456,raid1,raid0
```

Deze modules bevinden zich ook in de initrd.img (initiële ramdisk), dat het opstartstelsel is.

Op deze manier worden de raid-devices (/dev/mdx) herkend tijdens het booten.

```
yanu@kloon:~/eigen/linuxbabbel$ ls -l /initrd.img
lrwxrwxrwx 1 root root 28 2010-03-28 12:01 /initrd.img
-> boot/initrd.img-2.6.32-3-686
```

▶ **De SWAP-partitie:**

- ▶ enkel gebruikt wanneer de RAM volledig opgesoupeerd is,
- ▶ data niet belangrijk bij pech (ze wordt na het booten toch leeg gemaakt).

Voor een swap-partitie mag je zonder problemen **een RAID 0 partitie** aanmaken.

▶ **De home- en/of root-partitie:**

- ▶ dat heel belangrijk,
- ▶ veel lees- en schrijfoverdrachten,
- ▶ productiviteit belangrijk.

Als hier 1 schijf sneuvelt, kan je, zonder problemen, verder werken met de goeie schijf.

Vandaar dat **RAID 1 partities** hiervoor aangewezen zijn.

▶ Een data-partitie:

- ▶ opslag voor films, boeken, muziek, foto's, ...
- ▶ leesopdrachten zijn belangrijker dan schrijfopdrachten,

Hiervoor is een **RAID 5 partitie** interessant.

▶ Partitioneren van de schijven:

1. Je partitioneer 1 schijf,

Er bestaan verschillende programma's om dit te doen:
gparted, fdisk, cfdisk, ...

Opmerkingen:

- ▶ Vergeet de rootpartitie niet bootable te zetten.
- ▶ Dit is van groter belang!

Zet het **id-type** van alle partities op **Linux raid autodetect** of **fd**.

2. Je kopieert de partitie-tabel naar de andere schijven.

```
sfdisk -d /dev/sda | sfdisk /dev/sdb
```

Aanmaken van de RAID-arrays

Let op het raid-level 0 voor /dev/md1 en 5 voor /dev/md4.

▶ SWAP-partitie:

```
mdadm --create /dev/md1 -n 2 -l 0 /dev/sda1 /dev/sdb1
```

▶ De home- en root-partitie:

```
mdadm --create /dev/md2 -n 2 -l 1 /dev/sda2 /dev/sdb2
```

```
mdadm --create /dev/md3 -n 2 -l 1 /dev/sda3 /dev/sdb3
```

▶ De data-partitie:

```
mdadm --create /dev/md4 -n 3 -l 5 /dev/sdc1 /dev/sdd1 /dev/sde1
```

–*n* 2 betekent dat er twee disken gebruikt worden in die raid-array.

–*l* 0 betekent dat **raid-0** wordt gebruikt.

–*l* 1 betekent dat **raid-1** wordt gebruikt.

Aanmaken van de RAID-arrays

Wachten tot het syncen klaar is:

```
# watch cat /proc/mdstat
Personalities : [linear] [raid0] [raid1] [raid5] [multipath]
read_ahead 1024 sectors
md1 : active raid0 sdb1[1] sda1[0]
      995712 blocks 64k chunks
md2 : active raid1 sdb2[1] sda2[0]
      2000000 blocks [2/2] [UU]
md3 : active raid1 sdb3[2] sda3[0]
      75649984 blocks [2/1] [U_]
      [==>.....] recovery = 13.1% (9985448/75649984) finish=21.5mi
unused devices: <none>
```

Raid-Superblocks:

De superblocks moeten voor beiden overeenstemmende partities gelijk zijn.

De kernel zal ze detecteren en samenvoegen tot 1 raid-array (bv. /dev/md1).

Zijn de superblocks niet gelijk, dan kan je ze op 0 zetten:

```
# mdadm --zero-superblock /dev/sda1  
...
```

En de raid-arrays terug aanmaken (-A).

```
# mdadm -A /dev/md1 /dev/sda1 /dev/sdb1
```

De raid-arrays formateren met het gewenste filesystem (ext3, ext4, xfs, reiserfs, ...):

- ▶ De SWAP-partitie:

```
# mkswap /dev/md1
```

- ▶ De home- en root-partitie:

```
# mkfs.ext4 /dev/md2
```

```
# mkfs.ext4 /dev/md3
```

- ▶ De data-partitie:

```
# mkfs.ext4 /dev/md4
```

Installeren van het systeem

De partities zijn klaar:

- ▶ om data op te zetten,
- ▶ om een installatie te doen.

Opmerkingen:

- ▶ Raid-partities kunnen ook tijdens de installatieprocedure aangemaakt worden,
- ▶ Om dit alles te kunnen doen, heb je **een linuxsysteem** nodig, dit kan:
 - ▶ **een lifiesysteem** zijn (ubuntu-cd/stick, pxe-boot, . . .)
Dit is vooral nodig voor systeeminstallaties (home- en root-partitie).
 - ▶ **een bestaand systeem.**
Dit is perfect om voor extra data-partities.

De sheevaplug

Dit zal voor een volgende keer zijn ;)

