

# Korte uitleg: Disk partitionering

Disk wordt hier in zijn algemene vorm bedoeld, mass-storage dus. Dat kan een schijf zijn, maar ook een geheugenkaartje of een USB stick. Al dan niet verstopt in andere apparaten zoals camera's.

Hoewel wij nu vinden dat disks vroeger maar klein waren, was er al snel behoefte om een disk op te delen in stukken, die ieder voor zich weer als disk te gebruiken zouden moeten zijn. In Unix implementaties heten die stukken vaak "slices". Maar toen echte disks op PCs te beschikking kwamen werd daar binnen MS-DOS een implementatie voor gemaakt die de term partities gebruikt.

Voordelen:

- Verschillende partities kunnen verschillend gebruikt worden, bijvoorbeeld als swap of als file systeem of met file systemen van verschillend type zonder dat daar aparte disks voor nodig zijn.
- Verschillende delen kunnen van elkaar afgeschermd worden. Een gebruiker of een applicatie (denk aan een database) kan dan zijn eigen partitie geheel vullen zonder dat het systeem gedeelte daar last van heeft. Ook het gebruik van een aparte partitie voor `/home` zodat het systeem vervangen kan worden zonder invloed op de gebruikersgegevens valt hieronder.
- Het plaatsen van meer systemen, ieder in hun eigen partitie(s), hetgeen multi-boot mogelijk maakt.

Nadelen:

- De grenzen van een partitie zijn niet makkelijk aan te passen. Als er dus een partitie volloopt en er is in een andere nog ruimte, is dat frustrerend.

## MS-DOS/MBR partitionering

Omdat Linux vaak samen (multi-boot) met MS-DOS systemen op een PC wordt gebruikt moest Linux ook gebruik maken van dit type partitionering.

Toen de disk nog klein waren dacht men dat een mogelijkheid om die op te splitsen in maximaal vier (4) partities genoeg zou moeten zijn. daarvoor werd in het eerste blok van 512 bytes op de disk (het Master Boot Record, MBR) een tabel van vier maal 16 bits gemaakt (de rest van dit blok bevat de bootstrap code). Daarin staat voor ieder van die vier partities wat zijn begin en einde is (in Cylinder-Head-Sector), het partitie type en nog een paar dingen. Deze partities worden tegenwoordig de "primary partitions" genoemd.

Toen vier toch echt te weinig bleek is een truukje gevonden. Eén van de primary partitions bezet de hele rest van de disk. Deze kreeg de naam "extended partition". Meestal is dat de laatste partitie van een rijtje, maar niet noodzakelijk de vierde. Het is heel gewoon om primary partitions 1 en 2 te

hebben en dan extended partition 3. Aan het begin van de extended partition staan nu de gegevens van een deel van die extended partition. Dat noemen we een "logical partition". Uit die gegevens is ook op te maken waar de volgende logical partition beschreven wordt. Dat is dus een ketting. En een ketting heeft het voordeel dat hij erg lang kan worden. Er zijn dus veel logical partitions mogelijk (maar er zijn grenzen).

Samenvattend zijn er dus drie partition types:

- primary partitions, maximaal vier, genummers 1, 2, 3, 4;
- extended partition, maximaal één, dit is een primary partitie herkenbaar aan zijn partitietype;
- logical partitions, altijd genummerd 5 of hoger.

Sommige mensen raken in de war omdat de extended partitie veel ruimte inneemt waar op het eerste gezicht niets mee gebeurt. En dan zijn er nog andere partities die dezelfde ruimte gebruiken. Als je het bovenstaande begrijpt ben je dus weer gerustgesteld.

Een voorbeeld:

```
beneden:~ # fdisk -l
Disk /dev/sda: 200.0 GB, 200049647616 bytes
255 heads, 63 sectors/track, 24321 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x1c841c84

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1          1           574     4604008+   c   W95 FAT32 (LBA)
Partition 1 does not end on cylinder boundary.
/dev/sda2    *        575         8967     67416772+   7   HPFS/NTFS
/dev/sda3                8968        24321     123331005   f   W95 Ext'd (LBA)
/dev/sda5                8968         9229      2104483+   82  Linux swap / Solaris
/dev/sda6                9230        11840      20972826   83  Linux
/dev/sda7               11841        24321     100253601   83  Linux

beneden:~ #
```

Voor we bovenstaande lijst gaan bekijken eerst even iets over partitie types. Je ziet ze boven onder het kopje **Id**. Dat is een byte en wordt dus hexadecimaal getoond (00 - ff). **fdisk** is zo vriendelijk om daar ook een betekenis van de laten zien onder het kopje **System**. Je ziet daar ook wat er gebeurt als er geen centrale organisatie is die dit uitdeelt en registreert: dubbel bezettingen.

Deze types zijn alleen maar een byte. Het zegt niets over de werkelijke inhoud. Sommige programma's kijken er echter naar en het is het beste om ze goed te zetten.

- Partitie 1 is een restore partitie van het originele operating systeem en is erop gezet door de leverancier.
- Partitie 2 bevat een Windows XP systeem (en is dus geen HPFS, maar NTFS), dit Windows systeem kent deze partitie als de "c:" drive".
- Partitie 3 is de extended partitie (hier wordt het **Id f** zelfs door de BIOS

begrepen, moet dus goed zijn!), loopt tot het eind van de disk en omvat de volgende drie.

- Partitie 5 bevat de openSUSE swap partitie (en is dus Linux Swap en niet Solaris).
- Partitie 6 is de openSUSE root partitie.
- Partitie 7 is de openSUSE `/home` partitie.

We zien hier ook de namen van de device files die bij de disk en de partities horen. Zie daarvoor *Korte uitleg: Device files (/dev/sda en zo)*

Nog even iets over die partitietypes (`Id` in de `fdisk` uitvoer). Er zijn een aantal van deze types die beschreven worden als "Hidden ...." (bijvoorbeeld: `11 Hidden FAT12`). Dat gaat altijd over MicroSoft Windows filesysteemtypes en hun bestaan wordt waarschijnlijk door Windows verborgen gehouden voor Windows gebruiker (die moet je dom houden). Onder Linux is daar niets verborgens aan.

## GPT partitionering

Aangezien de disks de laatste jaren zeer groot zijn geworden is het MBR/MD-DOS schema geheel uit zijn mogelijkheden gebarsten. Het werd dus tijd voor iets nieuws: GPT.

GPT betekent GUID Partition Table, waarbij GUID betekent: Globally Unique Identifiers.

Uiteraard kan men hiermee zeer grote disks met veel partities aan. Hopelijk voor de verre toekomst. Helaas bevat deze standaard toch nog vage plekken en die worden door verschillende leveranciers dus verschillend gebruikt. In het algemeen (en al helemaal als je niet multi-boot en Linux dus alles laat regelen) gaat dit steeds beter. In het kader van een *Korte uitleg* gaan we niet op de details in, maar omdat er alleen maar "gewone" partities zijn is alles veel makkelijker te begrijpen.

Wel is het leuk te weten dat het eerste deel van een GPT vrij wordt gehouden. Dat eerste deel is zo groot als een MS-DOS/MBR tabel. Dat heeft twee gevolgen:

- als er per ongeluk een MS-DOS/MBR tabel naar de disk wordt geschreven is de GPT tabel nog steeds onbeschadigd;
- je kunt in dat stuk een MS-DOS/MBR tabel schrijven die zo goed mogelijk overeenkomt met de GPT tabel en dat geeft een beperkte opwaarde compatibiliteit.

Overigens is er aan het eind van de disk een kopie van de GPT tabel te vinden. Handig als de tabel onleesbaar is geworden.

## Partitioneren verplicht?

Nee, partitioneren is niet verplicht. Je kunt aan filesysteem net zo goed op een hele disk als op een partitie aanmaken. En hetzelfde geldt voor ander gebruik

van disk "containers" (Physical Volume van LVM, Swap, enz.) Als je dus geen reden ziet om de disk in meer dan één stuk op te delen is het niet nodig. Echter, omdat de partitietabel in het MBR zit, en je alleen kunt booten van een disk met een MBR, is een partitietabel verplicht bij een boot disk.