

Raspberry Pi: Hoe SD-kaarten te verkleinen

Met dank aan Jan, PA3EOZ (bron in Duitsland)

Ik (Erik, PA0EKA) liep al een tijd tegen dit euvel aan: ik wil op een SD-kaart geïnstalleerd Linux pakket overzetten naar een veel kleinere SD-kaart, omdat vaak nog geen kwart van de SD-ruimte wordt gebruikt. Een backup-kopie maken wordt dan ook veel kleiner ☺.

We moeten eerst een imagebestand van het SD-tje maken. Hiervoor gebruiken we Win32DiskImager in Windows of de loopback-functie onder Linux. In Linux: bij schijven -> schijfkopie aanmaken.

Dan wordt het nu spannend!

Om imagebestanden en de partities te verkleinen gebruiken we Linux en hebt u slechts drie tools nodig die al in de meeste Linux-distributies zijn opgenomen: 1. GParted, 2. Fdisk en 3. Truncate.

GParted kan alleen werken op fysieke apparaten, maar niet op images.

Als deze nog niet geactiveerd zijn, doen we dat met;

```
sudo modprobe loop
```

Nu kun je

```
sudo losetup -f
```

aanmaken met /dev/loop0-apparaat van het imagebestand.img:

```
sudo losetup /dev/loop0 Imagebestand.img
```

Het apparaat /dev/loop0 is beschikbaar, het bevat de totale inhoud van het imagebestand.img. We willen echter de afzonderlijke partities waarin de afzonderlijke partities staan bewerken. Dus we geven het als volgt aan de kernel vrij:

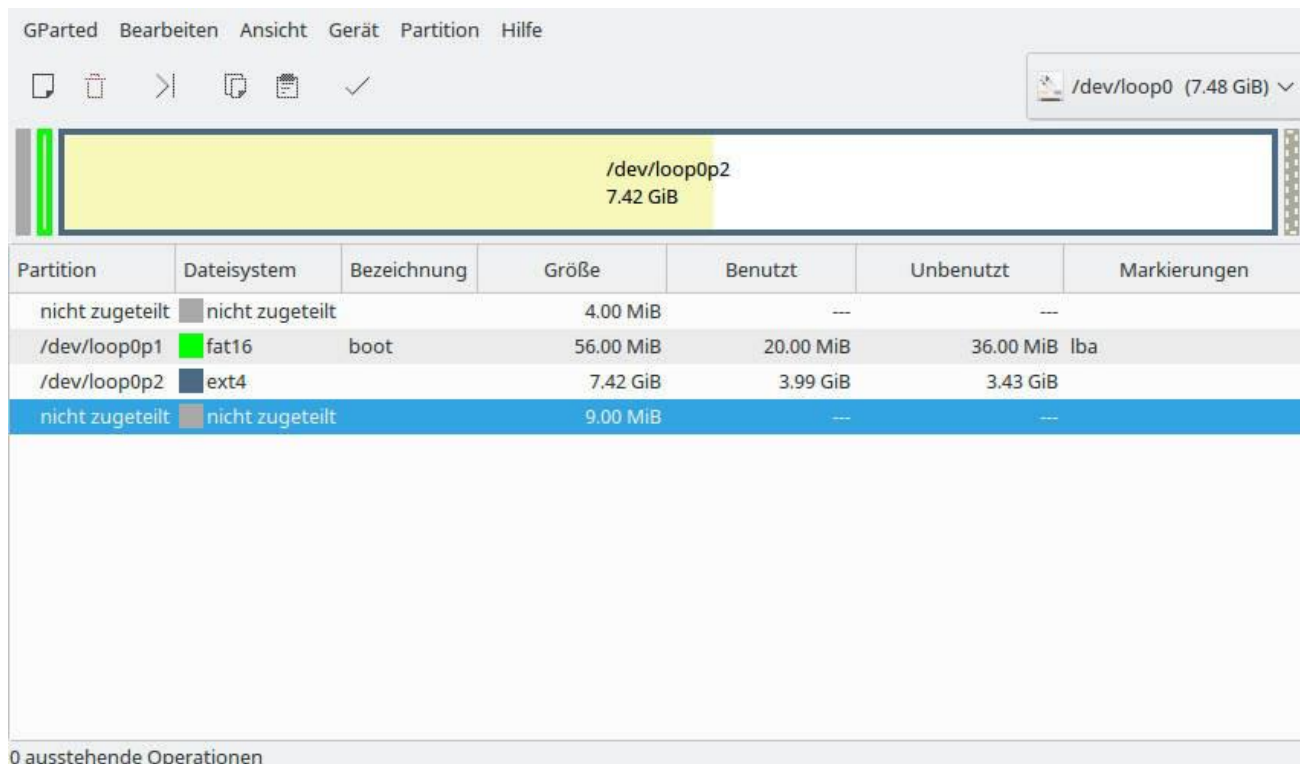
```
sudo partprobe /dev/loop0
```

Dit commando geeft ons **/dev/loop0p2** terug– de tweede beschikbare partitie in het imagebestand.img.

Dit is het apparaat waarmee GParted kan werken (de /dev/loop0p1 is de kleine FAT16-partitie die de Raspberry Pi moet opstarten - die we laten zoals hij is).

Nu staat het eindelijk op het punt om de partitie te verminderen. Hiervoor starten wij GParted als volgt:

```
sudo gparted /dev/loop0
```



Partition	Dateisystem	Bezeichnung	Größe	Benutzt	Unbenutzt	Markierungen
nicht zugeteilt	nicht zugeteilt		4.00 MiB	---	---	
/dev/loop0p1	fat16	boot	56.00 MiB	20.00 MiB	36.00 MiB	lba
/dev/loop0p2	ext4		7.42 GiB	3.99 GiB	3.43 GiB	
nicht zugeteilt	nicht zugeteilt		9.00 MiB	---	---	

0 ausstehende Operationen

Hier klikken we op de partitie **/dev/loop0p2** (Activeren) en vervolgens op de knop “Veranderen/Verbergen” (de knop die lijkt op een afspeelknop). In het venster dat nu verschijnt, zien we een balk die de grootte van de partitie en de daadwerkelijk gebruikte ruimte weergeeft. De rechterrand van de balk kan nu met de muis naar links worden gesleept – tot de gewenste kleinere partitie. Vervolgens klik je op “Veranderen/Shift”. Nu zijn we terug in het hoofdvenster (zie hierboven) en zie je: De partitie is nu kleiner en de nieuw verkregen vrije ruimte wordt weergegeven - althans dat is hoe het eruit ziet. Klik nu op het vinkje bovenaan de knoppenbalk – en afhankelijk van het aantal en soort bewerkingen, kan dit enige tijd gaan duren. Zodra deze procedure is afgerond, kunnen we GParted sluiten. We hebben het loopback-apparaat **/dev/loop0** niet meer nodig, zodat we het apparaat opnieuw vrijgeven

```
sudo losetup -d /dev/loop0
```

Nu is er al veel bereikt, maar als we met **ls -l** naar het **imagefile.img** kijken, zien we dat het nog steeds even groot is als voorheen. We moeten nog steeds de ruimte vrijmaken die vrijkomt door de partitie vermindering, zodat het **imagebestand.img** op een nieuwe, kleinere SD-kaart past. Om dit te doen, moeten we weten op welke geheugen positie de partitie begint en eindigt. Hiervoor gebruiken we **fdisk**:

```
sudo fdisk -l Imagebestand.img
```

Hier zien we enkele van de volgende:

```
Schijf /media/jan/Data/PI_Bullseye.img: 15,26 GiB, 16385048576 bytes, 32002048 sectoren
Eenheid: sectoren van 1 * 512 = 512 bytes
Sectorgrootte (logisch/fysiek): 512 bytes / 512 bytes
In-/uitvoergrootte (minimaal/optimaal): 512 bytes / 512 bytes
Schijflabeltype: dos
Schijf-ID: 0xdd95a7e9

Apparaat                               Op.   Begin   Einde  Sectoren  Grootte  ID  Type
/media/jan/Data/PI_Bullseye.img1       8192   93236   85045   41,5M    c  W95  FAT
/media/jan/Data/PI_Bullseye.img2       151552 29954047 29802496 14,2G    83  Linux
```

Hier besteden we aandacht aan het apparaat met de naam **Imagebestand.img2** in de laatste regel. In de image eindigt het in sector **29954047** (afhankelijk van hoeveel de verdeling in GParted is verminderd). Boven zien we dat elke sector 512 bytes is. Het voorbeeld in de afbeelding laat zien dat onze (verminderde) partitie eindigt op bytes **29954047 x 512**. Het enige dat erachter komt, is onze vrije en ongebruikte ruimte, waarmee we het **imagebestand.img** willen verkleinen.

Tot slot gebruiken we de tool **truncate**. Als parameter hebben we het eindblok van partitie 2 nodig in dit voorbeeld **29954047** en de blok grootte 512 bytes. In de praktijk ziet het commando er als volgt uit:

```
truncate --size=$((29954047+1)*512) Imagebestand.img
```

Nu is het doel bereikt: het beeldbestand is nu kleiner en we kunnen het **imagebestand.img** naar een nieuwe SD-kaart schrijven. Dit kan met **Disks** (Linux Mint) of **Win32 Disk Imager** (Windows).

Let op: geen schrijf/tikfouten maken !

Succes.