

# Resize a flash drive image

Reference: <https://askubuntu.com/questions/1174487/re-size-the-img-for-smaller-sd-card-how-to-shrink-a-bootable-sd-card-image>

Get a loop device:

Question: can I use the loop device that `mount -o loop` assigned when I mounted the image?  
Answer: you will need to unmount first, so you might need a new loop device. GParted can open a loop device before unmounting, but cannot resize a mounted partition. After unmounting, use `losetup` with an available loop device.

```
# get a loop device number
sudo losetup -f
# in this case: /dev/loop15
# attach the disk image to the loop device
sudo losetup /dev/loop15 test-disk.img
```

Use GParted with the device:

```
sudo gparted /dev/loop15
```

Right-click on the partition and select **Resize/Move**, and enter a new size.

Right-click and select **Information**. According to GParted (for this example), there is **980.47 MiB Unallocated** currently.

After completing resize, unload the loop device:

```
sudo losetup -d /dev/loop15
```

Use fdisk to get number of **sectors** and **unit** (or **block**) size:

```
fdisk -l test-disk.img
```

Disk /dev/loop15: 29.28 GiB, 31436800000 bytes, 61400000 sectors

Units: sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disklabel type: dos

Disk identifier: 0x00000000

61400000\*512 bytes to GiB = 29.2778015 gibibytes

61400000\*512 bytes to MiB = 29 980.4688 mebibytes

So we need to truncate up to 980 MiB (in this case, from GParted info) from the end of the file.

60000000\*512 bytes to MiB = 29 296.875 mebibytes, which should work well in this case.

Now we use truncate to shorten the file:

```
truncate --size=$(( 60000000+1 ) * 512) test-disk.img
```

Now, might want to check/repair with GParted:

```
sudo losetup /dev/loop15 test-disk.img  
sudo gparted /dev/loop15
```

---

Revision #1

Created 10 April 2024 18:32:19 by giw

Updated 10 April 2024 19:21:55 by giw