



Exercise 17.3 Using losetup and parted

We are going to experiment more with:

- Loop devices and **losetup**
- **parted** to partition at the command line non-interactively.

We expect that you should read the **man pages** for **losetup** and **parted** before doing the following procedures.

Once again, you can reuse the image file or, better still, zero it out and start freshly or with another file.

1. Associate the image file with a **loop** device:

```
$ sudo losetup -f
/dev/loop1
$ sudo losetup /dev/loop1 imagefile
```

where the first command finds the first **free** loop device. The reason to do this is you may already be using one or more loop devices. For example, on the system that this is being written on, before the above command is executed:

```
$ losetup -a
/dev/loop0: []: (/usr/src/KERNELS.sqfs)
```

a **squashfs** compressed, read-only filesystem is already mounted using **/dev/loop0**. (The output of this command will vary with distribution.) If we were to ignore this and use **losetup** on **/dev/loop0** we would almost definitely corrupt the file.

2. Create a disk partition label on the loop device (image file):

```
$ sudo parted -s /dev/loop1 mklabel msdos
```

3. Create three primary partitions on the loop device:

```
$ sudo parted -s /dev/loop1 unit MB mkpart primary ext4 0 256
$ sudo parted -s /dev/loop1 unit MB mkpart primary ext4 256 512
$ sudo parted -s /dev/loop1 unit MB mkpart primary ext4 512 1024
```

4. Check the partition table:

```
$ fdisk -l /dev/loop1
Disk /dev/loop1: 1073 MB, 1073741824 bytes, 2097152 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
Disk label type: dos
Disk identifier: 0x00050c11

    Device Boot      Start         End      Blocks   Id  System
/dev/loop1p1        1         500000       250000    83  Linux
/dev/loop1p2     500001     1000000       250000    83  Linux
/dev/loop1p3     1000001     2000000       500000    83  Linux
```

5. What happens next depends on what distribution you are on. For example, on **RHEL 7** and **Ubuntu 16.04** you will find new device nodes have been created:

```
$ ls -l /dev/loop1*
brw-rw---- 1 root disk  7, 1 Oct  7 14:54 /dev/loop1
brw-rw---- 1 root disk 259, 0 Oct  7 14:54 /dev/loop1p1
brw-rw---- 1 root disk 259, 3 Oct  7 14:54 /dev/loop1p2
brw-rw---- 1 root disk 259, 4 Oct  7 14:54 /dev/loop1p3
```

and we will use them in the following.

6. Put filesystems on the partitions:

```
$ sudo mkfs.ext3 /dev/loop1p1
$ sudo mkfs.ext4 /dev/loop1p2
$ sudo mkfs.vfat /dev/loop1p3
```

7. Mount all three filesystems and show they are available:

```
$ mkdir mnt1 mnt2 mnt3

$ sudo mount /dev/loop1p1 mnt1
$ sudo mount /dev/loop1p2 mnt2
$ sudo mount /dev/loop1p3 mnt3

$ df -Th
Filesystem                Type      Size  Used Avail Use% Mounted on
/dev/sda1                  ext4       29G   8.5G   19G   32% /
....
/dev/loop1p1               ext3     233M   2.1M   219M    1% mnt1
/dev/loop1p2               ext4     233M   2.1M   215M    1% mnt2
/dev/loop1p3               vfat     489M     0   489M    0% mnt3
```

8. After using the filesystems to your heart's content you can unwind it all:

```
$ sudo umount mnt1 mnt2 mnt3
$ rmdir mnt1 mnt2 mnt3
$ sudo losetup -d /dev/loop1
```