

# RHSA 2 - LAB 4

1. Use fdisk -l to locate information about the partition sizes.

```
smabrouk@localhost:~$ sudo fdisk -l
Disk /dev/sda: 119.2 GiB, 128835676160 bytes, 250069600 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: gpt
Disk identifier: 8676880E-8820-43FE-AFA2-7708E694E2D0

Device            Start      End      Sectors  Size Type
/dev/sda1         2048      616447    614400    300M EFI System
/dev/sda2         616448    876591    262144    128M Microsoft reserved
/dev/sda3         876592   145825791 144947200 69.1G Microsoft basic data
/dev/sda4        248225792 250868991  2643200    900M Windows recovery environment
/dev/sda5        145825792 147922943  2097152    1G Linux filesystem
/dev/sda6        147922944 150820892  2897152    1G Linux filesystem
/dev/sda7        150820896 246854911  96034816 45.8G Linux LVM

Partition table entries are not in disk order.

Disk /dev/sdb: 931.5 GiB, 1000264888000 bytes, 1953525168 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: gpt
Disk identifier: 018C2D84-1B17-4ABF-95EC-26312D1E0534

Device            Start      End      Sectors  Size Type
/dev/sdb1         2048    620795903 620793856 296G Microsoft basic data
/dev/sdb2         620795904 1235195903 614400000 293G Microsoft basic data
/dev/sdb3        1917134848 1953523711  36388864 17.4G Windows recovery environment

Disk /dev/mapper/rhel-root: 25 GiB, 26843545600 bytes, 52428800 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 /dev/mapper/rhel-swap: 4.0 GiB, 5138022400 bytes, 10035200 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 /dev/mapper/rhel-home: 16 GiB, 17179869184 bytes, 33554432 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 /dev/loop0: 89.5 MiB, 93818880 bytes, 183240 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 /dev/loop1: 91.7 MiB, 96141512 bytes, 187776 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 /dev/loop2: 372.4 MiB, 390451200 bytes, 762600 sectors
```

2. Use fdisk to add a new logical partition that is 1GB in size.

```
smabrouk@localhost:~$ sudo fdisk /dev/mapper/rhel-root

Welcome to fdisk (util-linux 2.32.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

The old xfs signature will be removed by a write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xe24d17eb.

Command (m for help): m

Help:
DOS (MBR)
a toggle a bootable flag
b edit nested BSD disklabel
c toggle the dos compatibility flag

Generic
d delete a partition
l list known partition types
n add a new partition
p print the partition table
t change a partition type
v verify the partition table
i print information about a partition

Misc
m print this menu
u change display/entry units
x extra functionality (experts only)

Script
I load disk layout from sfdisk script file
O dump disk layout to sfdisk script file

Save & Exit
w write table to disk and exit
q quit without saving changes

Create a new label
g create a new empty GPT partition table
G create a new empty SGI (IRIX) partition table
o create a new empty DOS partition table
s create a new empty Sun partition table

Command (m for help): v
Warning: 512000 unallocated 512 byte sectors.

Command (m for help): n
Partition type:
p primary (0 primary, 0 extended, 4 free)
e extended (container for logical partitions)
Select (default p):
```

3. Did the kernel feel the changes? Display the content of /proc/partitions file? What did you notice? How to overcome that?

```
smabrouk@localhost:~  
File Edit View Search Terminal Help  
Salma >> cat /proc/partitions  
major minor #blocks name  
8 0 125034840 sda  
8 1 307200 sda1  
8 2 131072 sda2  
8 3 72473600 sda3  
8 4 921600 sda4  
8 5 1048576 sda5  
8 6 1048576 sda6  
8 7 48017408 sda7  
8 16 976762584 sdb  
8 17 310396928 sdb1  
8 18 307200000 sdb2  
8 19 18194432 sdb3  
11 0 1048575 sr0  
253 0 26214400 dm-0  
253 1 5017600 dm-1  
253 2 16777216 dm-2  
7 0 91620 loop0  
7 1 93888 loop1  
7 2 381300 loop2  
7 3 56948 loop3  
7 4 64748 loop4  
7 5 4 loop5  
7 6 50812 loop6  
7 7 354640 loop7  
7 8 64760 loop8  
Salma >> █
```

4. Make a new ext2 file system on the new logical partition you just created. Bonus: Try creating the ext2 filesystem with 2k blocks and one inode per every 4k (two blocks) of filesystem.

```
smabrouk@localhost:~  
File Edit View Search Terminal Help  
Salma >> mkfs.ext4 /dev/sdb1  
mke2fs 1.45.6 (20-Mar-2020)  
Could not open /dev/sdb1: Permission denied  
Salma >> sudo mkfs.ext4 /dev/sdb1  
mke2fs 1.45.6 (20-Mar-2020)  
/dev/sdb1 contains a nfs file system labelled 'Others'  
Proceed anyway? (y,N) y  
Creating filesystem with 77599232 4k blocks and 19406848 inodes  
Filesystem UUID: a5990991-33a0-47e0-b536-263caa3129b8  
Superblock backups stored on blocks:  
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,  
4096000, 7962624, 11239424, 20480000, 23887872, 71663616  
Allocating group tables: done  
Writing inode tables: done  
Creating journal (262144 blocks): done  
Writing superblocks and filesystem accounting information: done  
Salma >> █
```

5. Create a directory, name it /data.

```
smabrouk@localhost:~  
File Edit View Search Terminal Help  
Salma >> sudo mkdir /mount_point_for_sdb1  
Salma >> sudo mount -t ext4 /dev/sdb1 /mount_point_for_sdb1  
Salma >> df -h /mount_point_for_sdb1/  
Filesystem Size Used Avail Use% Mounted on  
/dev/sdb1 291G 28K 276G 1% /mount_point_for_sdb1  
Salma >> sudo mkdir /mount_point_for_sdb1/data  
Salma >> █
```

6. Add a label to the new filesystem, name it data.

```
smabrouk@localhost:~  
File Edit View Search Terminal Help  
Salma >> sudo e2label /dev/sdb1 /data  
Salma >> █
```

7. Add a new entry to /etc/fstab for the new filesystem using the label you just create.

```
smabrouk@localhost:~  
File Edit View Search Terminal Help  
#  
# /etc/fstab  
# Created by anaconda on Tue Nov 22 12:13:10 2022  
#  
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.  
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.  
#  
# After editing this file, run 'systemctl daemon-reload' to update systemd  
# units generated from this file.  
#  
/dev/mapper/rhel-root / xfs defaults 0 0  
UUID=b338a207-9604-44ca-9290-038dbf090f2b /boot xfs defaults 0 0  
/dev/mapper/rhel-home /boot/efi vfat umask=0077,shortname=winnt 0 2  
/dev/mapper/rhel-home /home xfs defaults 0 0  
/dev/mapper/rhel-swap none swap defaults 0 0  
LABEL=data /data ext4 defaults 0 0  
█
```

## 8. Mount the new filesystem.

```
smabrouk@localhost:~  
File Edit View Search Terminal Help  
Salma >> mount /dev/sdb1 /data
```

## 9. Display your swap size.

```
smabrouk@localhost:~  
File Edit View Search Terminal Help  
Salma >> swapon -s  
Filename                                Type      Size    Used    Priority  
/dev/dm-1                               partition 5017596 0       -2  
Salma >> █
```

## 10. Create a swap file of size 512MB.

```
smabrouk@localhost:~  
File Edit View Search Terminal Help  
Salma >> sudo vi /etc/fstab  
Salma >> sudo swapon -a  
swapon: cannot find the device for LABEL=SWAP-sdb1  
Salma >> █
```