13 – Storage Management

Basic Commands for Storage Partitions

fdisk

- fdisk is used to check the partitions on a disk.
- The fdisk command can display the partitions and details like file system type.

```
cbkpc@Ubuntu22:~$ sudo fdisk -l
[sudo] password for cbkpc:
Disk /dev/loop0: 61.89 MiB, 64901120 bytes, 126760 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: 4 KiB, 4096 bytes, 8 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: 63.45 MiB, 66535424 bytes, 129952 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
```

sfdisk utility purpose similar to fdisk, but with more features.

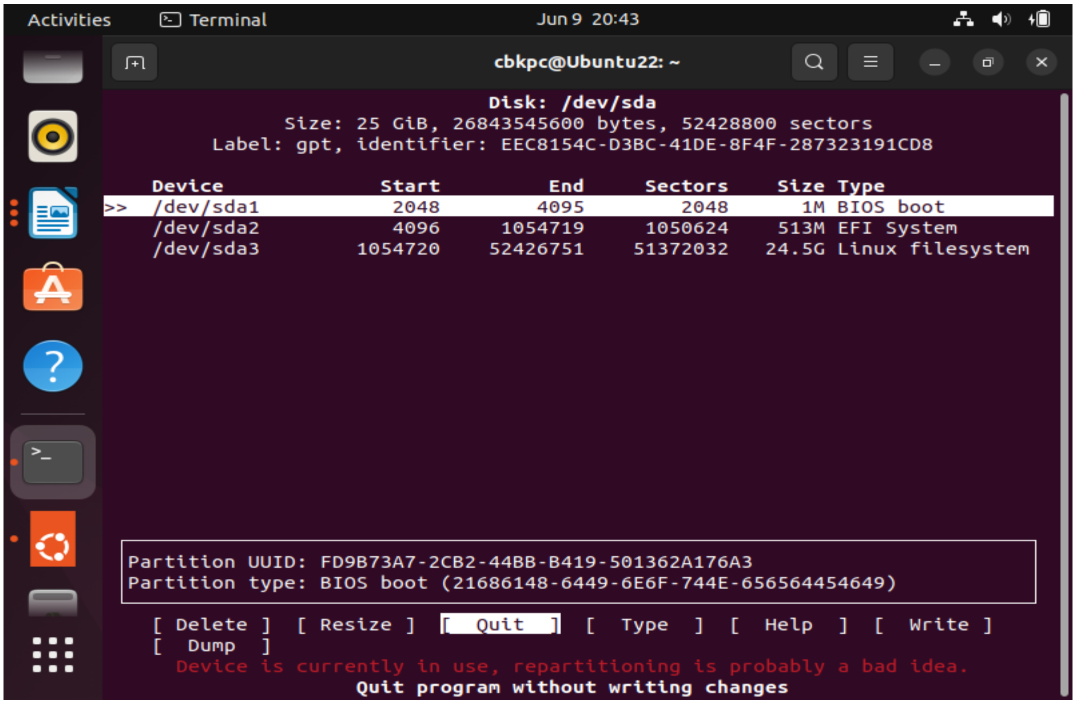
sfdisk

```
cbkpc@Ubuntu22:~$ sudo sfdisk -l
Disk /dev/loop0: 61.89 MiB, 64901120 bytes, 126760 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: 4 KiB, 4096 bytes, 8 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
```

cfdisk

- cfdisk is a linux partition editor with an interactive user interface based on **ncurses**.
- It can be used to list out the existing partitions as well as create or modify them.



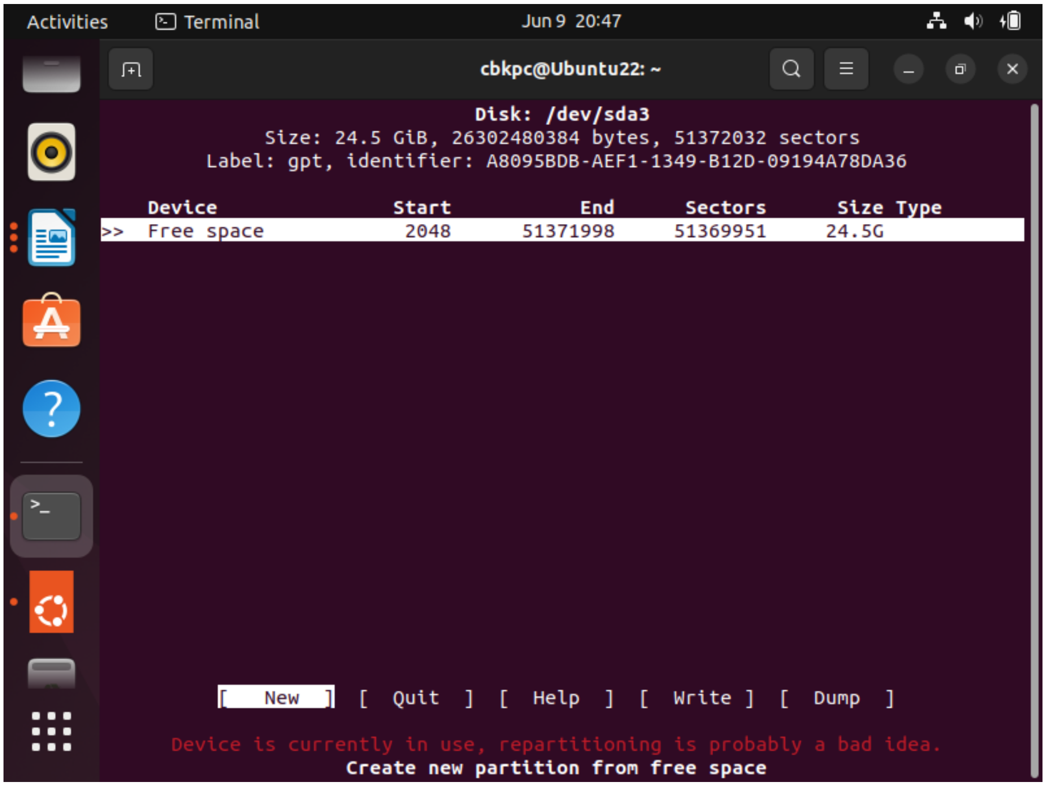
```
                              Disk: /dev/sda
                 Size: 25 GiB, 26843545600 bytes, 52428800 sectors
           Label: gpt, identifier: EEC8154C-D3BC-41DE-8F4F-287323191CD8

    Device                   Start         End     Sectors     Size Type
>>  /dev/sda1                 2048        4095        2048       1M BIOS boot
    /dev/sda2                 4096     1054719     1050624     513M EFI System
    /dev/sda3              1054720    52426751    51372032    24.5G Linux filesystem

 Partition UUID: FD9B73A7-2CB2-44BB-B419-501362A176A3
 Partition type: BIOS boot (21686148-6449-6E6F-744E-656564454649)

    [ Delete ]  [ Resize ]  [  Quit  ]  [  Type  ]  [  Help  ]  [  Write ]
    [  Dump  ]
      Device is currently in use, repartitioning is probably a bad idea.
                     Quit program without writing changes
```

cbkpc@Ubuntu22:~$ sudo cfdisk /dev/sda3



```
                              Disk: /dev/sda3
                Size: 24.5 GiB, 26302480384 bytes, 51372032 sectors
           Label: gpt, identifier: A8095BDB-AEF1-1349-B12D-09194A78DA36

    Device                   Start          End     Sectors     Size Type
>>  Free space                2048     51371998    51369951    24.5G

          [  New   ]  [  Quit  ]  [  Help  ]  [  Write ]  [  Dump  ]

      Device is currently in use, repartitioning is probably a bad idea.
                      Create new partition from free space
```

parted

- partedutility is to list out partitions and modify them if needed.

```
cbkpc@Ubuntu22:~$ sudo parted –l

Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sda: 26.8GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name                  Flags
 1      1049kB  2097kB  1049kB                                     bios_grub
 2      2097kB  540MB   538MB   fat32        EFI System Partition  boot, esp
 3      540MB   26.8GB  26.3GB  ext4
```

df

- df is not a partitioning utility, but prints out details about only mounted file systems.
- The list generated by df even includes file systems that are not real disk partitions.
    - **df -h**

```
cbkpc@Ubuntu22:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           233M  1.6M  231M   1% /run
/dev/sda3        24G   11G   12G  48% /
tmpfs           1.2G     0  1.2G   0% /dev/shm
tmpfs           5.0M  4.0K  5.0M   1% /run/lock
/dev/sda2       512M  5.3M  507M   2% /boot/efi
tmpfs           233M  152K  232M   1% /run/user/1000
```

- **Note:**df shows only the mounted file systems or partitions and not all.

pydf

- pydf is an improved version of df, written in python.
- Prints out all the hard disk partitions in a easy to read manner.

lsblk

- Lists out all the storage blocks, which includes disk partitions and optical drives.
- Details include the total size of the partition/block and the mount point if any.

```
cbkpc@Ubuntu22:~$ lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
loop0    7:0    0  61.9M  1 loop /snap/core20/1405
loop1    7:1    0     4K  1 loop /snap/bare/5
loop2    7:2    0  63.5M  1 loop /snap/core20/1891
loop3    7:3    0 155.6M  1 loop /snap/firefox/1232
loop4    7:4    0  73.8M  1 loop /snap/core22/750
loop5    7:5    0 248.8M  1 loop /snap/gnome-3-38-2004/99
loop6    7:6    0 349.7M  1 loop /snap/gnome-3-38-2004/140
```

```
loop7    7:7    0 460.6M  1 loop /snap/gnome-42-2204/102
loop8    7:8    0 460.7M  1 loop /snap/gnome-42-2204/105
loop9    7:9    0  81.3M  1 loop /snap/gtk-common-themes/1534
loop10   7:10   0  45.9M  1 loop /snap/snap-store/575
loop11   7:11   0  12.3M  1 loop /snap/snap-store/959
loop12   7:12   0  91.7M  1 loop /snap/gtk-common-themes/1535
loop13   7:13   0  53.3M  1 loop /snap/snapd/19361
loop14   7:14   0   284K  1 loop /snap/snapd-desktop-integration/10
loop15   7:15   0   452K  1 loop /snap/snapd-desktop-integration/83
sda      8:0    0    25G  0 disk
├─sda1   8:1    0     1M  0 part
├─sda2   8:2    0   513M  0 part /boot/efi
└─sda3   8:3    0  24.5G  0 part /
sr0     11:0    1  1024M  0 rom
```

blkid

- Prints the block device (partitions and storage media) attributes like uuid and file system type. Does not report the space on the partitions.

cbkpc@Ubuntu22:~$ blkid

```
/dev/sda3: UUID="ea4ba3fc-f651-4ffa-a84c-fec055d923f3" BLOCK_SIZE="4096"
TYPE="ext4" PARTUUID="6f6e0051-4b82-438c-8975-6fde7db81de3"
/dev/loop1: TYPE="squashfs"
/dev/loop8: TYPE="squashfs"
/dev/loop15: TYPE="squashfs"
/dev/loop6: TYPE="squashfs"
/dev/loop13: TYPE="squashfs"
/dev/loop4: TYPE="squashfs"
/dev/loop11: TYPE="squashfs"
/dev/loop2: TYPE="squashfs"
/dev/loop0: TYPE="squashfs"
/dev/loop9: TYPE="squashfs"
/dev/loop7: TYPE="squashfs"
/dev/sda2: UUID="B82A-5E06" BLOCK_SIZE="512" TYPE="vfat" PARTLABEL="EFI
System Partition" PARTUUID="7fda5595-7569-452d-9c38-b404173541f7"
/dev/loop14: TYPE="squashfs"
/dev/loop5: TYPE="squashfs"
/dev/loop12: TYPE="squashfs"
/dev/loop3: TYPE="squashfs"
/dev/loop10: TYPE="squashfs"
```

hwinfo

- The hwinfo is a general purpose hardware information tool and can be used to print out the disk and partition list.
- The output however does not print details about each partition like the above commands.
    - **hwinfo --block --short**

inxi

- inxi command display information about various hardware components present on the system.
- To display information about the disk drives and storage devices use the "-D" option with inxi.
    - **inxi -D -xx**

=== * ===

Logical Volume Management (LVM)

- **LVM**, or Logical Volume Management, is a storage device management technology that gives users the power to pool and abstract the physical layout of component storage devices for easier and flexible administration.
- The main advantages of LVM are increased abstraction, flexibility, and control.
- Logical volumes can have meaningful names like “databases” or “root-backup”.
- Volumes can be resized dynamically as space requirements change and migrated between physical devices within the pool on a running system or exported easily.
- LVM also offers advanced features like snapshotting, striping, and mirroring.

LVM Storage Management Structures

LVM functions by layering abstractions on top of physical storage devices. The basic layers that LVM uses, starting with the most primitive, are.

- **Physical Volumes**:
    - **LVM utility prefix**: pv...
    - **Description**: Physical block devices or other disk-like devices (for example, other devices created by device mapper, like RAID arrays) are used by LVM as the raw building material for higher levels of abstraction. Physical volumes are regular storage devices. LVM writes a header to the device to allocate it for management.

- **Volume Groups**:
    - **LVM utility prefix**: vg...
    - **Description**: LVM combines physical volumes into storage pools known as volume groups. Volume groups abstract the characteristics of the underlying devices and function as a unified logical device with combined storage capacity of the component physical volumes.
- **Logical Volumes**:
    - **LVM utility prefix**: lv... (generic LVM utilities might begin with lvm...)
    - **Description**: A volume group can be sliced up into any number of logical volumes. Logical volumes are functionally equivalent to partitions on a physical disk, but with much more flexibility. Logical volumes are the primary component that users and applications will interact with.

Each volume within a volume group is segmented into small, fixed-size chunks called **extents**. The size of the extents is determined by the volume group (all volumes within the group conform to the same extent size).

=== * ===