

# TP n°1 GNU/Linux

## Partie 1 :

- 1- La commande qui nous permet d'afficher le partitionnement du disque est `fdisk -l`

Mais lorsque on execute cette commande on reçoit le message « permission denied » :

```
nour@nour-Vostro-3500:~$ man fdisk
nour@nour-Vostro-3500:~$ fdisk -l
fdisk: cannot open /dev/loop0: Permission denied
fdisk: cannot open /dev/loop1: Permission denied
fdisk: cannot open /dev/loop2: Permission denied
fdisk: cannot open /dev/loop4: Permission denied
fdisk: cannot open /dev/loop5: Permission denied
fdisk: cannot open /dev/loop6: Permission denied
fdisk: cannot open /dev/loop7: Permission denied
fdisk: cannot open /dev/sda: Permission denied
fdisk: cannot open /dev/loop8: Permission denied
fdisk: cannot open /dev/loop10: Permission denied
fdisk: cannot open /dev/loop9: Permission denied
fdisk: cannot open /dev/loop11: Permission denied
fdisk: cannot open /dev/loop12: Permission denied
fdisk: cannot open /dev/loop13: Permission denied
fdisk: cannot open /dev/loop14: Permission denied
fdisk: cannot open /dev/loop15: Permission denied
fdisk: cannot open /dev/loop16: Permission denied
fdisk: cannot open /dev/loop18: Permission denied
fdisk: cannot open /dev/loop17: Permission denied
fdisk: cannot open /dev/loop19: Permission denied
fdisk: cannot open /dev/loop20: Permission denied
fdisk: cannot open /dev/loop21: Permission denied
fdisk: cannot open /dev/loop22: Permission denied
fdisk: cannot open /dev/loop23: Permission denied
fdisk: cannot open /dev/loop24: Permission denied
fdisk: cannot open /dev/loop26: Permission denied
fdisk: cannot open /dev/loop27: Permission denied
```

Alors on a conclu que on a besoin des privilège d'un « super user » pour afficher la liste des partitions du disque .

On exécutant la commande `sudo fdisk -l` et après l'authentification

La liste des partitions du disque s'affiche :

```
nour@nour-Vostro-3500:~$ sudo fdisk -l
[sudo] password for nour:
Disk /dev/loop0: 146.97 MiB, 154083328 bytes, 300944 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: 72.92 MiB, 76443648 bytes, 149304 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/loop4: 116.71 MiB, 122363904 bytes, 238992 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/loop5: 375.7 MiB, 393289728 bytes, 768144 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/loop6: 243.8 MiB, 254889984 bytes, 497832 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/loop7: 55.58 MiB, 57196544 bytes, 111928 sectors
```

- 2- Pour connaître la taille totale du disque on a utilisé la commande `df -h` l'option `-h` affiche les tailles des partitions d'une manière lisible pour l'humain ( en Go,Mo,...) puis on a sommé les tailles des partitions pour avoir la taille total du disque :

```
nour@nour-Vostro-3500:~$ df -hT
Filesystem      Type      Size  Used Avail Use% Mounted on
udev            devtmpfs  1.8G   0    1.8G   0% /dev
tmpfs           tmpfs     366M   2.2M 364M   1% /run
/dev/sda3       ext4      908G   41G   821G   5% /
tmpfs           tmpfs     1.8G  129M   1.7G   8% /dev/shm
tmpfs           tmpfs     5.0M   4.0K  5.0M   1% /run/lock
tmpfs           tmpfs     1.8G   0    1.8G   0% /sys/fs/cgroup
/dev/loop1      squashfs  128K   128K   0 100% /snap/bare/5
/dev/loop0      squashfs  147M   147M   0 100% /snap/chromium/2254
/dev/loop8      squashfs   64M    64M   0 100% /snap/core20/1738
/dev/loop11     squashfs   64M    64M   0 100% /snap/core20/1778
/dev/loop6      squashfs  244M   244M   0 100% /snap/code/116
/dev/loop9      squashfs  117M   117M   0 100% /snap/core/14399
/dev/loop15     squashfs   56M    56M   0 100% /snap/cups/872
/dev/loop13     squashfs   73M    73M   0 100% /snap/core22/469
/dev/loop10     squashfs   56M    56M   0 100% /snap/core18/2654
/dev/loop22     squashfs  219M   219M   0 100% /snap/gnome-3-34-1804/72
/dev/loop16     squashfs   92M    92M   0 100% /snap/gtk-common-themes/1535
/dev/loop14     squashfs   56M    56M   0 100% /snap/cups/836
/dev/loop21     squashfs  165M   165M   0 100% /snap/gnome-3-28-1804/161
/dev/loop19     squashfs  347M   347M   0 100% /snap/gnome-3-38-2004/115
/dev/loop18     squashfs  219M   219M   0 100% /snap/gnome-3-34-1804/77
/dev/loop24     squashfs   46M    46M   0 100% /snap/snap-store/599
/dev/loop17     squashfs   82M    82M   0 100% /snap/gtk-common-themes/1534
/dev/loop7      squashfs   56M    56M   0 100% /snap/core18/2667
/dev/loop20     squashfs  347M   347M   0 100% /snap/gnome-3-38-2004/119
/dev/loop26     squashfs   46M    46M   0 100% /snap/snap-store/638
/dev/loop27     squashfs  370M   370M   0 100% /snap/telegram-desktop/4384
/dev/sda1       vfat      871M   32M   840M   4% /boot/efi
tmpfs           tmpfs     366M   48K   366M   1% /run/user/1001
/dev/loop2      squashfs   73M    73M   0 100% /snap/core22/484
/dev/loop4      squashfs  117M   117M   0 100% /snap/core/14447
/dev/loop12     squashfs  248M   248M   0 100% /snap/code/117
/dev/loop5      squashfs  376M   376M   0 100% /snap/telegram-desktop/4486
/dev/loop23     squashfs  149M   149M   0 100% /snap/chromium/2271
```

Alors la taille totale du disque est : 918.591 Go

3- On des différents systèmes de gestion de fichiers utilisées par les différentes partitions du disque :

Les partitions loop utilisent **squashfs** :

- Squashfs est un système de fichiers compressés utilisé pour la lecture seulement pour les systèmes sous unix .

Les partitions tmpfs utilisent **tmpfs** (temporary file system) :

- Tmpfs contient les fichiers qui se suppriment a l'arrêt de système.

La partition sda3 qui est la plus grande partition utilise **ext4** :

- Ext4 (extended file system 4) est le successeur de ext3  
Utilisé par les systèmes unix pour la gestion des fichiers

⇒ D'autres systèmes de gestion de fichiers qu'on connaît sont :

- **FAT** (file allocation table) la taille maximale d'un fichier que Fat peut manipuler est 2Go et ne réalise pas la gestion des droits d'accès.
- **BTRFS** (B-TREE file system ) il réalise une meilleur gestion d'espace pour les fichiers de petite taille et

s'occupe de la gestion des droit  
d'accès

4- `/dev/hda1` c'est un répertoire contenant les données stockées sur la première partition du premier disque dur IDE

`/dev/hdb3` c'est un répertoire contenant les données stockées sur la troisième partition du deuxième disque dur IDE

`/dev/sdc` c'est un répertoire contenant les données stockées sur la troisième disque dur SATA

5- Le GNU GRand Boot loader (ou GRUB) est un programme permettant à l'utilisateur de sélectionner le système d'exploitation ou noyau qui doit être chargé au démarrage du système. Il permet également à l'utilisateur de transmettre des arguments au noyau.

Le fichier GRUB se trouve dans `/boot` :

```
nour@nour-Vostro-3500:/$ cd boot
nour@nour-Vostro-3500:/boot$ ls
config-5.14.0-1054-oem  initrd.img          memtest86+.bin      System.map-5.14.0-1056-oem  vmlinuz.old
config-5.14.0-1056-oem  initrd.img-5.14.0-1054-oem  memtest86+.elf      vmlinuz
efi                   initrd.img-5.14.0-1056-oem  memtest86+_multiboot.bin  vmlinuz-5.14.0-1054-oem
grub                  initrd.img.old        System.map-5.14.0-1054-oem  vmlinuz-5.14.0-1056-oem
```

Le fichier de configuration du GRUB `grub.cfg` se trouve dans `/boot/grub` :

```
grub
nour@nour-Vostro-3500:/boot$ cd grub
nour@nour-Vostro-3500:/boot/grub$ ls
fonts  grub.cfg  grubenv  unicode.pf2  x86_64-efi
nour@nour-Vostro-3500:/boot/grub$
```

⇒ L'option default spécifie le système qui se boot par défaut lors du démarrage.

⇒ L'option timeout spécifie en secondes la durée après laquelle GRUB ne charge plus l'entrée spécifiée dans l'option default.

## 6- La structure de /

```
nour@nour-Vostro-3500:/boot/grub$ cd /  
nour@nour-Vostro-3500:/ $ ls  
bin  cdrom  etc  lib  lib64  lost+found  mnt  proc  run  snap  swapfile  tmp  var  
boot dev  home lib32 libx32 media  opt  root  sbin  srv  sys  usr
```

**/home** : contient les dossiers des utilisateurs

**/dev** : contient les fichiers liés aux périphériques de machine

**/boot** : contient les fichiers de démarrage de linux

**/etc** : contient les fichiers de configuration de système de des applications.

**/lib** : Les librairies et bibliothèques partagés pour le fonctionnement de l'OS et des applications

**/mnt** : Les répertoires utilisés pour monter temporairement un système de fichiers.

**/proc** : Répertoire virtuel avec les informations système (l'état du système) basé sur procfs (*process file system*)

## Partie 2 :

- 7- L'affichage du contenu de la variable d'environnement `PATH` se fait à travers la commande **`echo $PATH`**. Le résultat affiché est une liste de répertoires que le système parcourt à la recherche d'exécutables quand l'utilisateur tape une commande. Les répertoires sont séparés par `' : '`.

```
yasmine@yas-VirtualBox:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin
```

- 8- La commande **`echo $SHELL`** permet d'afficher le chemin du shell actuel. Sur Ubuntu, le chemin par défaut est `/bin/bash`.

```
yasmine@yas-VirtualBox:~$ echo $SHELL
/bin/bash
```

- 9- Les commandes **`ls`**, **`useradd`** et **`chsh`** se trouvent dans le répertoire `/usr/bin`. **`ls`** permet de lister les fichiers et les sous-répertoires dans un répertoire, **`useradd`** permet de créer un nouveau compte utilisateur et **`chsh`** est utilisée pour changer le shell login d'un utilisateur.
- 10- La création des deux groupes nécessite les privilèges d'un `'superuser'` ce qui fait appel à l'ajout du mot clé **`sudo`** au début de la commande. L'accès n'est acquis qu'après la saisie d'un mot de passe correcte du `'superuser'`

```
yasmine@yas-VirtualBox:~$ sudo groupadd gr1 gr2
[sudo] password for yasmine:
Usage: groupadd [options] GROUP
```

Vérification du placement des groupes dans le répertoire **`/etc/group`** :

```
yasmine@yas-VirtualBox:~$ grep 'gr1' /etc/group
gr1:x:1001:user1
yasmine@yas-VirtualBox:~$ grep 'gr2' /etc/group
gr2:x:1002:user2
```

- 11- gr1 et gr2 sont de GID 1001 et 1002 respectivement.
- 12- Création de ''user1'' dans ''gr1'' et ''user2'' dans ''gr2'' avec l'ajout du mot clé **sudo** :

- L'option -m sert à créer un nouveau répertoire pour l'utilisateur
- L'option -G spécifie le(s) groupe(s) au(x)quel(s) le nouvel utilisateur appartient

```
yasmine@yas-VirtualBox:~$ sudo useradd -m -G gr1 user1
[sudo] password for yasmine:
yasmine@yas-VirtualBox:~$ sudo useradd -m -G gr2 user2
```

- 13- Pour l'attribution du mot de passe, on a toujours besoin de l'accès privilégié

```
yasmine@yas-VirtualBox:~$ sudo passwd user1
[sudo] password for yasmine:
New password:
Retype new password:
passwd: password updated successfully
```

- Fixation de la validité du mot de passe :
- M** indique le nombre maximal de jours pendant lesquels l'utilisateur peut bénéficier du mot de passe

```
yasmine@yas-VirtualBox:~$ sudo chage -M 5 user1
yasmine@yas-VirtualBox:~$ sudo chage -M 5 user2
```

- 14- Vérification de l'emplacement des utilisateurs :

```
yasmine@yas-VirtualBox:~$ grep 'user1' /etc/passwd
user1:x:1001:1003::/home/user1:/bin/sh
yasmine@yas-VirtualBox:~$ grep 'user2' /etc/passwd
user2:x:1002:1004::/home/user2:/bin/sh
yasmine@yas-VirtualBox:~$
```

- Pour ''user1'' et ''user2'', les UID sont 1001 et 1002 (resp) et se trouvent dans le troisième champ, les GID sont 1003 et 1004 (resp) et se trouvent dans le quatrième champ. Le dernier champ est réservé pour le shell /bin/bash pour les deux utilisateurs.
- Le deuxième champ est réservé pour le mot de passe représenté par le caractère ''x''.



15-

```
yasmine@yas-VirtualBox:~$ sudo grep 'user1' /etc/shadow
[sudo] password for yasmine:
user1:$y$j9T$IN44uLR2LNagIS7oom.6c1$eJgWUR76G14aIB1CxZaglCgki
I5Xxg4Td.xgc7BLFc0:19381:0:5:7:::

yasmine@yas-VirtualBox:~$ sudo grep 'user2' /etc/shadow
user2:$y$j9T$4NDwL52M3T5EV6.KYDamu1$FCgDAjIci5NZuCbyuXsS/Xdmk
raJABX00/EluX.AckA:19381:0:5:7:::
```

Le deuxième champ est occupé par la version hachée du mot de passe de l'utilisateur sous forme d'une chaîne de caractères pour des raisons de sécurité. Et ce hash ne peut pas être décrypté.

16-

- La connexion en tant que `'user1'` ou `'user2'` se fait à travers la commande **su**

```
yasmine@yas-VirtualBox:~$ su user1
Password:
Warning: your password will expire in 5 days.
$
$ pwd
/home/yasmine
```

- L'accès au répertoire personnel de l'utilisateur se fait par la commande **cd /home/user1** ou **cd ~**

```
$ cd /home/user1
$ pwd
/home/user1
```

17- Pour accéder au répertoire racine, on utilise la commande **cd /**. Et **cd ..** pour retourner au répertoire précédent.

18- Les commandes **su** ou **sudo -i** permettent la connexion en tant que `'root'`

- Pour changer le groupe auquel appartient l'utilisateur, on a recours à la commande **usermod -g gr2 user1**

```
usermod: cannot open /etc/passwd: try again later
yasmine@yas-VirtualBox:~$ sudo usermod -g gr2 user1
[sudo] password for yasmine:
yasmine@yas-VirtualBox:~$
```

Vérification du nouvel emplacement par la commande **id user1**

```
yasmine@yas-VirtualBox:~$ id user1
uid=1001(user1) gid=1002(gr2) groups=1002(gr2),1001(gr1)
```

- 19- On peut changer le shell de user2 à bash grâce à la commande ***usermod -s /bin/bash user2***

```
usermod: cannot lock /etc/passwd, try again later.  
yasmine@yas-VirtualBox:~$ sudo usermod -s /bin/bash user2
```

- Vérification de la modification par deux méthodes :

```
yasmine@yas-VirtualBox:~$ grep user2 /etc/passwd  
user2:x:1002:1004:~/home/user2:/bin/bash  
yasmine@yas-VirtualBox:~$ cat /etc/passwd | grep user2  
user2:x:1002:1004:~/home/user2:/bin/bash  
yasmine@yas-VirtualBox:~$
```