

Lab 3

2.

Question 1

```
student@vm04:~/lab3$ touch test.txt
student@vm04:~/lab3$ ls -l
total 0
-rw-r--r-- 1 student student 0 Oct 22 20:53 test.txt
```

On a le jeu de permissions suivant :

- Propriétaire : lecture et écriture
- Groupe : lecture
- Autres : lecture

Question 2

```
student@vm04:~/lab3$ ls -la
total 12
drwxr-xr-x 3 student student 4096 Oct 23 12:51 .
drwx----- 7 student student 4096 Oct 22 20:52 ..
drwxr-xr-x 2 student student 4096 Oct 23 12:51 question2
```

On remarque l'ajout du d pour directory (répertoire) et le jeu de permissions suivant :

- Propriétaire : lecture, écriture et exécution
- Groupe : lecture et exécution
- Autres : lecture et exécution

3.

Question 3

```
student@vm04:~$ cat /etc/group | grep student
cdrom:x:24:student
floppy:x:25:student
sudo:x:27:student
audio:x:29:student
dip:x:30:student
video:x:44:student
plugdev:x:46:student
users:x:100:student
netdev:x:106:student
student:x:1000:-
```

En utilisant la commande « cat /etc/group » on peut voir la liste de tous les groupes existants.

Puis on remarque que l'unique utilisateur possible dans les différentes listes d'utilisateurs est « student ». Donc on utilise la commande « cat /etc/group | grep student » pour repérer toutes les lignes où on a un utilisateur « student ». Néanmoins, il faut faire attention au groupe student qui ne contient aucune liste d'utilisateurs. Cela nous permet d'avoir la liste complète des groupes qui ont une liste d'utilisateurs non vide.

Liste des groupes et leurs id :

cdrom 24
floppy 25
sudo 27
audio 29
dip 30
video 44
plugdev 46
users 100
netdev 106

Question 4

```
student@vm04:~$ id securite
uid=1001(securite) gid=1001(securite) groups=1001(securite)
```

L'Identifiant de l'utilisateur securite est 1001 et il appartient uniquement au groupe securite.

Question 5

```
student@vm04:~$ sudo groupadd unine^C
student@vm04:~$ cat /etc/group | grep unine
unine:x:1002:
```

L'identifiant du nouveau groupe unine est 1002.

Question 6

```
student@vm04:~$ sudo usermod -aG unine student
student@vm04:~$ sudo usermod -aG unine securite
student@vm04:~$ id securite
uid=1001(securite) gid=1001(securite) groups=1001(securite),1002(unine)
student@vm04:~$ id student
uid=1000(student) gid=1000(student) groups=1000(student),24(cdrom),25(floppy),27(sudo),29(audio),30(dip),44(video),46(plugdev),100(users),106(netdev),1002(unine)
student@vm04:~$ cat /etc/group | grep unine
unine:x:1002:student,securite
```

On voit bien sur ce Screenshot que tout marche parfaitement.

La commande utilisée « sudo usermode -aG ... » est la même que celle proposée, c'est simplement une manière plus courte de l'écrire.

4.

Question 7

- r-x = 5
- rw- = 6
- rwx = 7
- --- = 0
- --x = 1

Question 8

- -rw-r--r-- = 644
- drwxr-xr-x = 755
- -rw----- = 600
- -r---x-w- = 412
- dr-x-w--w- = 522

Question 9

```
student@vm04:~/lab3$ touch test.txt
student@vm04:~/lab3$ ls -l
total 0
-rw-r--r-- 1 student student 0 Oct 22 20:53 test.txt
```

(screenshot repris de l'exercice 1)

Le propriétaire du fichier est student et le groupe est également student.

Question 10

```
student@vm04:~/lab3$ chmod 660 test.txt
student@vm04:~/lab3$ ls -l
total 4
drwxr-xr-x 2 student student 4096 Oct 23 12:51 question2
-rw-rw---- 1 student student 0 Oct 22 20:53 test.txt
student@vm04:~/lab3$ █
```

On remarque bien que les permissions du fichier test.txt ont changé. La valeur numérique utilisée est 660.

Question 11

```
student@vm04:~/lab3$ chmod o+r test.txt
student@vm04:~/lab3$ ls -l
total 4
drwxr-xr-x 2 student student 4096 Oct 23 12:51 question2
-rw-rw-r-- 1 student student  0 Oct 22 20:53 test.txt
student@vm04:~/lab3$
```

L'abréviation utilisée est o+r.

Question 12

```
student@vm04:~/lab3$ ls -l
total 4
drwxr-xr-x 2 student student 4096 Oct 23 12:51 question2
-rw-rw-r-- 1 student student  0 Oct 22 20:53 test.txt
student@vm04:~/lab3$ umask 013
student@vm04:~/lab3$ umask -S
U=rwx,g=rw,o=r
student@vm04:~/lab3$ touch question12.txt
student@vm04:~/lab3$ ls -l
total 4
-rw-rw-r-- 1 student student  0 Oct 23 15:23 question12.txt
drwxr-xr-x 2 student student 4096 Oct 23 12:51 question2
-rw-rw-r-- 1 student student  0 Oct 22 20:53 test.txt
student@vm04:~/lab3$
```

La valeur numérique utilisée est 013. On peut confirmer que c'est bien juste en utilisant la commande « umask -S » qui nous permet d'avoir le détail sur les permissions par défaut pour les différentes classes (propriétaire, groupe, autres). Ensuite, lors de la création du fichier question12.txt, on constate que les permissions du fichier sont sous la forme suivante : -rw-rw-r--, c'est le résultat attendu, car par défaut l'exécution est désactivée lors de la création de fichiers (on ne peut pas contourner cette règle avec umask).

5.

Question 13

```
student@vm04:~/lab3$ ls -l
total 4
-rw-rw-r-- 1 student student  0 Oct 23 15:23 question12.txt
-rw-r----- 1 securite unine  0 Oct 23 15:46 question13
drwxr-xr-x 2 student student 4096 Oct 23 12:51 question2
-rw-rw-r-- 1 student student  0 Oct 22 20:53 test.txt
student@vm04:~/lab3$
```

Les changements ont bien été effectués. Les commandes utilisées sont, dans l'ordre :

1. Touch question13
2. Chmod 640 question13
3. Sudo chown securite:unine question13

```
student@vm04:~/lab3$ cat question13
slt
student@vm04:~/lab3$ echo "add" >> question13
-bash: question13: Permission denied
student@vm04:~/lab3$
```

```
GNU nano 7.2 question13 *
slt salut tout le monde j'essaie de modifier le fichier

[ Error writing question13: Permission denied ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
```

Il est possible de l'ouvrir et de lire le contenu du fichier. Cependant, il n'est pas possible de le modifier. Cela est dû au fait que nous faisons partie du groupe unine donc nous avons uniquement le droit à la lecture du fichier. Cependant, nous ne sommes plus propriétaire du fichier ici le propriétaire est désormais securite et donc les droits de propriétaire nous ont été enlevé.