

SHELL

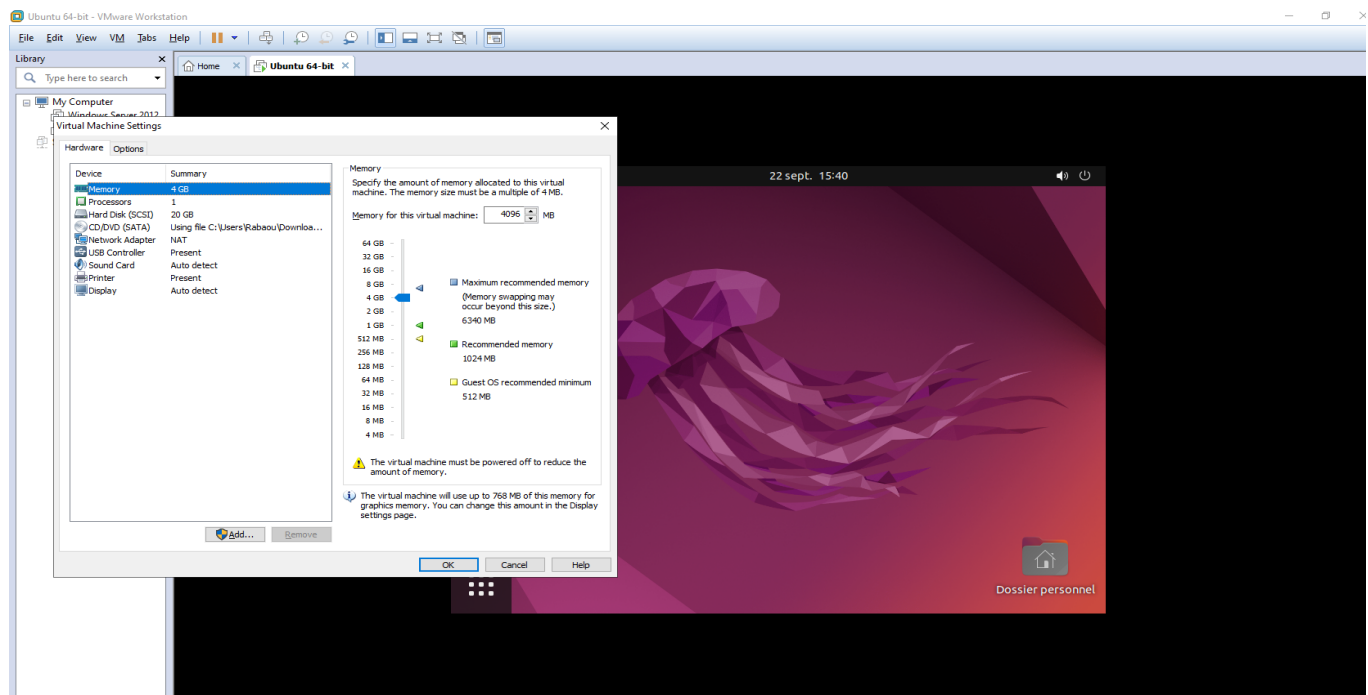
Après avoir rapidement appris quelques bases pour se familiariser avec le terminal lors du premier projet ENIAC, nous revoilà à travailler de nouveau dans un terminal.

Le terminal dans lequel ce projet a été réalisé est celui de Ubuntu, le système d'exploitation utilisé dans le logiciel VMWare permettant de créer des machines virtuelles.

Configuration de la machine virtuelle Ubuntu :

- 4 Go de RAM
- 20 Go alloués en espace disque

Pour chaque question j'essaierai d'ajouter une image pour montrer mes résultats.



JOB 01

Le but du JOB 01 consiste à approfondir nos connaissances sur le terminal à travers l'apprentissage de nouvelles commandes ainsi que des fonctionnalités que l'on peut ajouter.

- Afficher le manuel de la commande ls

La commande "man" suivie de la commande dont veut connaître les détails nous donne le résultat que l'on souhaite avoir.

En tapant donc la commande : "man ls", on obtient la page manuel de la commande ls comme on peut le voir sur cette capture d'écran :

```
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.
```

```
rabaou@rabaou-virtual-machine:~$ man ls  
rabaou@rabaou-virtual-machine:~$
```

```
LS(1)                                User Commands                                LS(1)

NAME
  ls - list directory contents

SYNOPSIS
  ls [OPTION]... [FILE]...

DESCRIPTION
  List information about the FILES (the current directory by default).  
Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-  
fied.

  Mandatory arguments to long options are mandatory for short options  
too.

  -a, --all
      do not ignore entries starting with .

  -A, --almost-all
      do not list implied . and ..

  --author
      with -l, print the author of each file

  -b, --escape
      print C-style escapes for nongraphic characters

Manual page ls(1) line 1 (press h for help or q to quit)
```

- Afficher les fichiers cachés du home de votre utilisateur

La commande “ls -a” sert à lister **tous** les éléments de home dont ceux qui sont cachés (d’où le “-a” qui signifie “all”).

```
rabaou@rabaou-virtual-machine:~$ ls -a
.          Bureau      .gnupg      Modèles     Public      .thunderbird
..         .cache      Images      .mozilla    snap        Vidéos
.bash_logout .config    .lessht     Musique     .ssh
.bashrc     Documents  .local      .profile    Téléchargements
rabaou@rabaou-virtual-machine:~$
```

- Afficher les fichiers cachés plus les informations sur les droits sous forme de liste

La commande pour lister les droits des fichiers d’un répertoire est “ls -l”. On peut ajouter “-a” que l’on a utilisé plus haut pour regarder les droits concernant les fichiers cachés comme on le voit sur cette capture.

```
rabaou@rabaou-virtual-machine:~$ ls -a -l
total 88
drwxr-x--- 18 rabaou rabaou 4096 sept. 19 19:11 .
drwxr-xr-x  3 root  root  4096 sept. 19 18:38 ..
-rw-r--r--  1 rabaou rabaou  220 sept. 19 18:38 .bash_logout
-rw-r--r--  1 rabaou rabaou 3771 sept. 19 18:38 .bashrc
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Bureau
drwx----- 14 rabaou rabaou 4096 sept. 19 19:04 .cache
drwx----- 10 rabaou rabaou 4096 sept. 19 18:57 .config
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Documents
drwx-----  2 rabaou rabaou 4096 sept. 19 18:57 .gnupg
drwxr-xr-x  3 rabaou rabaou 4096 sept. 19 19:00 Images
-rw-----  1 rabaou rabaou   20 sept. 19 19:11 .lessht
drwx-----  3 rabaou rabaou 4096 sept. 19 18:55 .local
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Modèles
drwx-----  3 rabaou rabaou 4096 sept. 19 19:01 .mozilla
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Musique
-rw-r--r--  1 rabaou rabaou  807 sept. 19 18:38 .profile
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Public
drwx-----  4 rabaou rabaou 4096 sept. 19 19:03 snap
drwx-----  2 rabaou rabaou 4096 sept. 19 18:57 .ssh
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Téléchargements
drwx-----  6 rabaou rabaou 4096 sept. 19 19:01 .thunderbird
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Vidéos
```

Maintenant comment comprendre ces lignes ?

Pour chaque ligne, la série de symboles donne comme informations la nature du fichier concerné.

Si la série commence par “-” il s’agit d’un fichier, “l” pour un lien et “d” pour un répertoire.

Puis on a 3 symboles à 3 reprises, cela correspond aux droits du propriétaire, du groupe puis des utilisateurs. Les lettres comme “r” “w” ou “x ” sont utilisées pour montrer ce que l’on peut faire des fichiers en question.:

- r correspond à read (lire)
- w correspond à write (écrire)
- x correspond à execute (exécuter)
- “-” est noté lorsque qu’il n’y a pas de permission.

- Comment ajouter des options à une commande ?

Les commandes du terminal ont généralement des options que l’on ajoute après avoir tapé la commande en question. Par exemple “ls -a” ou “ls -l”

```
rabaou@rabaou-virtual-machine:~$ man ls
rabaou@rabaou-virtual-machine:~$ ls -a
.          Bureau  .gnupg    Modèles   Public     .thunderbird
..         .cache  Images    .mozilla  snap       Vidéos
.bash_logout .config .lessht    Musique   .ssh
.bashrc     Documents .local     .profile  Téléchargements
rabaou@rabaou-virtual-machine:~$
```

- Quelles sont les deux syntaxes principales d'écriture des options pour une commande ?

Quand on écrit une commande et qu’on lui ajoute une option, l’ordre donné dans les options n’a pas d’importance si l’on veut les regrouper. Par exemple si on veut lister les droits des fichiers cachés on ajoutera comme option après la commande “ls” -a et -l
On se rend compte que l’on peut taper “ls -l -a” comme on peut taper “ls -la” ou “ls -al” pour arriver au même résultat.

```

rabaou@rabaou-virtual-machine:~$ ls -al
total 88
drwxr-x--- 18 rabaou rabaou 4096 sept. 19 19:11 .
drwxr-xr-x  3 root   root   4096 sept. 19 18:38 ..
-rw-r--r--  1 rabaou rabaou  220 sept. 19 18:38 .bash_logout
-rw-r--r--  1 rabaou rabaou 3771 sept. 19 18:38 .bashrc
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Bureau
drwx----- 14 rabaou rabaou 4096 sept. 19 19:04 .cache
drwx----- 10 rabaou rabaou 4096 sept. 19 18:57 .config
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Documents
drwx-----  2 rabaou rabaou 4096 sept. 19 18:57 .gnupg
drwxr-xr-x  3 rabaou rabaou 4096 sept. 19 19:00 Images
-rw-----  1 rabaou rabaou   20 sept. 19 19:11 .lessht
drwx-----  3 rabaou rabaou 4096 sept. 19 18:55 .local
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Modèles
drwx-----  3 rabaou rabaou 4096 sept. 19 19:01 .mozilla
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Musique
-rw-r--r--  1 rabaou rabaou  807 sept. 19 18:38 .profile
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Public
drwx-----  4 rabaou rabaou 4096 sept. 19 19:03 snap
drwx-----  2 rabaou rabaou 4096 sept. 19 18:57 .ssh
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Téléchargements
drwx-----  6 rabaou rabaou 4096 sept. 19 19:01 .thunderbird
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Vidéos
rabaou@rabaou-virtual-machine:~$

```

```

rabaou@rabaou-virtual-machine:~$ ls -la
total 88
drwxr-x--- 18 rabaou rabaou 4096 sept. 19 19:11 .
drwxr-xr-x  3 root   root   4096 sept. 19 18:38 ..
-rw-r--r--  1 rabaou rabaou  220 sept. 19 18:38 .bash_logout
-rw-r--r--  1 rabaou rabaou 3771 sept. 19 18:38 .bashrc
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Bureau
drwx----- 14 rabaou rabaou 4096 sept. 19 19:04 .cache
drwx----- 10 rabaou rabaou 4096 sept. 19 18:57 .config
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Documents
drwx-----  2 rabaou rabaou 4096 sept. 19 18:57 .gnupg
drwxr-xr-x  3 rabaou rabaou 4096 sept. 19 19:00 Images
-rw-----  1 rabaou rabaou   20 sept. 19 19:11 .lessht
drwx-----  3 rabaou rabaou 4096 sept. 19 18:55 .local
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Modèles
drwx-----  3 rabaou rabaou 4096 sept. 19 19:01 .mozilla
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Musique
-rw-r--r--  1 rabaou rabaou  807 sept. 19 18:38 .profile
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Public
drwx-----  4 rabaou rabaou 4096 sept. 19 19:03 snap
drwx-----  2 rabaou rabaou 4096 sept. 19 18:57 .ssh
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Téléchargements
drwx-----  6 rabaou rabaou 4096 sept. 19 19:01 .thunderbird
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Vidéos

```

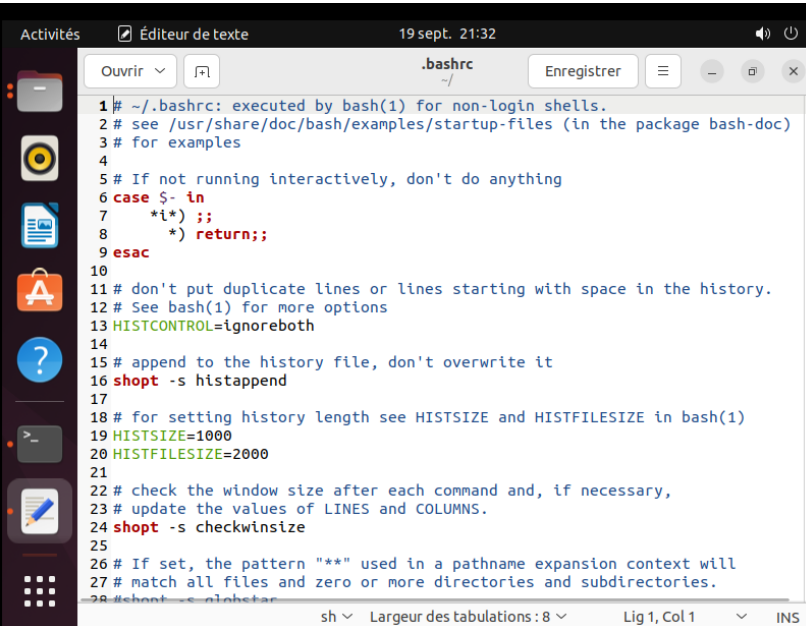
JOB 02

Il faut tout d'abord trouver le fichier ".bashrc" sur lequel on va travailler. Pour ce faire, il faut d'abord l'afficher pour ensuite examiner les changements qu'on fera au fur et à mesure via les commandes.

La commande pour ouvrir un fichier est : "open "nom du fichier""

On exécute la commande (voir première capture et voilà le résultat (voir deuxième capture) :

```
rabaou@rabaou-virtual-machine:~$ open .bashrc
```



The screenshot shows a terminal window titled "Éditeur de texte" (Text Editor) with the file ".bashrc" open. The window has a dark theme and a sidebar on the left with various application icons. The main area displays the content of the ".bashrc" file, which includes comments and configuration for the bash shell. The status bar at the bottom indicates "sh", "Largeur des tabulations: 8", "Lig 1, Col 1", and "INS".

```
1 # ~/.bashrc: executed by bash(1) for non-login shells.
2 # see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
3 # for examples
4
5 # If not running interactively, don't do anything
6 case $- in
7   *) ;;
8   *) return;;
9 esac
10
11 # don't put duplicate lines or lines starting with space in the history.
12 # See bash(1) for more options
13 HISTCONTROL=ignoreboth
14
15 # append to the history file, don't overwrite it
16 shopt -s histappend
17
18 # for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
19 HISTSIZE=1000
20 HISTFILESIZE=2000
21
22 # check the window size after each command and, if necessary,
23 # update the values of LINES and COLUMNS.
24 shopt -s checkwinsize
25
26 # If set, the pattern "*" used in a pathname expansion context will
27 # match all files and zero or more directories and subdirectories.
28 #shopt -s globstar
```

Maintenant que c'est fait on va pouvoir tenter de réaliser toutes les opérations qu'on nous demande :

- Lisez un fichier en utilisant une commande qui permet seulement de lire

Pour afficher le contenu d'un fichier on utilise la commande "cat".
On met donc la commande "cat .bashrc". Vu que le contenu du fichier est très long, on ne mettra pas d'images de captures d'écran.

- Afficher les 10 premières lignes du fichier ".bashrc"

Pour afficher la première partie d'un fichier (tout en indiquant le nombre de lignes en option) on utilise la commande "head".

Donc on entre la commande : "head -n 10 .bashrc"

```
rabaou@rabaou-virtual-machine:~$ head -n 10 .bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

rabaou@rabaou-virtual-machine:~$
```

- Afficher les 10 dernières lignes du fichier ".bashrc"

Pour afficher les 10 dernières lignes du fichier (tout en indiquant le nombre de lignes en option) on utilise la commande "tail". Donc on tape au clavier la commande : "tail -n 10 .bashrc". On obtient ce résultat :

```
rabaou@rabaou-virtual-machine:~$ tail -n10 .bashrc
# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi

rabaou@rabaou-virtual-machine:~$
```


- Afficher les 20 premières lignes du fichier “.bashrc”

Par déduction, on utilise la commande “head”, suivie de “-n 20”. Donc on entre la commande : “head -n 20 .bashrc” :

```
rabaou@rabaou-virtual-machine:~$ head -n20 .bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000
rabaou@rabaou-virtual-machine:~$
```

- Afficher les 20 dernières lignes du fichier “.bashrc”

Ici, on utilise toujours la commande “tail”, suivie de “-n 20” pour indiquer qu’on veut afficher les 20 dernières lignes.

Donc on entre la commande : “tail -n 20 .bashrc” :

```
rabaou@rabaou-virtual-machine:~$ tail -n20 .bashrc
# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi
rabaou@rabaou-virtual-machine:~$
```


JOB 03

Je n'ai pas pu faire l'intégralité du JOB 03, car je n'ai pas accès à internet dans mon VMWare.

Je me connecte à internet dans le système d'exploitation hôte avec l'application NETGEAR et une clé Wifi. En revanche, je n'ai pas trouvé de moyen sur Ubuntu pour utiliser cette clé WiFi.

Du coup je ne peux pas installer le paquet "cmatrix" dans mon ordinateur.

En revanche j'ai essayé de répondre aux questions pour montrer ce qu'il faut faire dans chaque cas :

- Installer le paquet "cmatrix"

Lorsque nous sommes sur Ubuntu, les logiciels se composent d'un ou plusieurs paquets.

Pour installer un paquet, on tape la commande suivante :

"sudo apt install mon_paquet".

Donc ici on remplace "mon_paquet" par "cmatrix" pour ensuite taper :

"sudo apt-get install cmatrix"

- Lancer le paquet que vous venez d'installer

Pour ouvrir un fichier .exe, il faut d'abord taper la commande cd pour aller vers le répertoire où le fichier exécutable se trouve. Ensuite on tape la commande "wine nom_du_fichier".

- Mettre à jour son gestionnaire de paquets

Si l'on veut maintenant réaliser la mise à jour du gestionnaire de packages, on utilise la commande : "sudo apt-get update"

```
rabaou@rabaou-virtual-machine:~$ sudo apt-get update
Ign :1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign :2 http://fr.archive.ubuntu.com/ubuntu jammy InRelease
Ign :3 http://fr.archive.ubuntu.com/ubuntu jammy-updates InRelease
Ign :4 http://fr.archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign :1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign :2 http://fr.archive.ubuntu.com/ubuntu jammy InRelease
Ign :3 http://fr.archive.ubuntu.com/ubuntu jammy-updates InRelease
Ign :4 http://fr.archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign :1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign :2 http://fr.archive.ubuntu.com/ubuntu jammy InRelease
Ign :3 http://fr.archive.ubuntu.com/ubuntu jammy-updates InRelease
Ign :4 http://fr.archive.ubuntu.com/ubuntu jammy-backports InRelease
```

- Mettre à jour ses différents logiciels

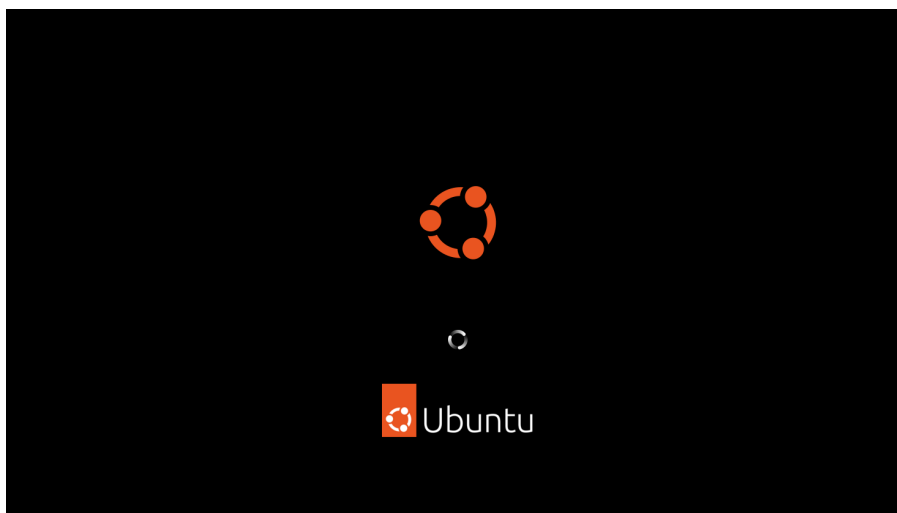
Pour la MAJ des logiciels, on tape : “sudo apt-get upgrade”.

```
rabaou@rabaou-virtual-machine:~$ sudo apt-get upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
rabaou@rabaou-virtual-machine:~$
```

- Redémarrer votre machine

Pour redémarrer la machine, on tape la commande “sudo reboot”

```
rabaou@rabaou-virtual-machine:~$ sudo reboot
[sudo] Mot de passe de rabaou : █
```



- Télécharger les internets : Google

Pour télécharger un fichier sur internet on peut utiliser la commande wget, on tape "wget" avec l'adresse du site.

- Éteindre votre machine

Pour l'éteindre, on tape la commande "shutdown -h".

```
rabaou@rabaou-virtual-machine:~$ shutdown -h
Shutdown scheduled for Fri 2022-09-23 14:13:07 CEST, use 'shutdown -c' to cancel.
rabaou@rabaou-virtual-machine:~$ █
```

JOB 04

Pour créer un fichier on tape la commande “cat” suivi du symbole “>” avec ensuite le nom du fichier.txt. On peut ensuite taper sur le terminal le contenu que l’on veut ajouter dans le document texte. Lorsque l’on a fini, on tape Ctrl+D pour enregistrer.

Exemple :

```
rabaou@rabaou-virtual-machine:~$ cat > users.txt
User 1
User 2
```

Ensuite pour créer un groupe il faut utiliser la commande “groupadd” suivi du nom du groupe.

Si je tape “groupadd Plateformeurs” on me donnera pas la permission de créer ce groupe.

```
rabaou@rabaou-virtual-machine:~$ groupadd Plateformeurs
groupadd: Permission denied.
groupadd : impossible de verrouiller /etc/group ; veuillez réessayer plus tard.
```

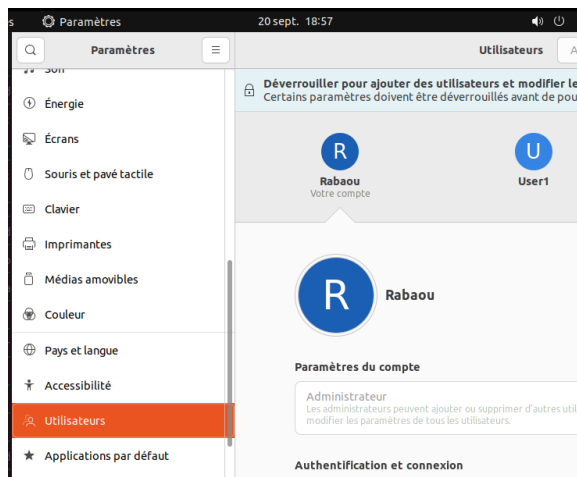
Je peux par contre le faire en tapant la commande “sudo groupadd Plateformeurs”.

```
rabaou@rabaou-virtual-machine:~$ sudo groupadd Plateformeurs
```

On va s’inspirer de cette commande pour créer les deux utilisateurs :

```
rabaou@rabaou-virtual-machine:~$ sudo useradd User1
```

On peut voir dans les paramètres que User1 existe :



(En revanche comme on peut le voir l’affichage de l’écran est tronqué donc on ne peut pas vérifier pour User2).

- Ajouter “User2” au groupe Plateformeurs

En gros, pour ajouter un utilisateur vers un groupe il faut taper la commande :

“sudo usermod -a -G “nom du groupe” + “nom de l’utilisateur””.

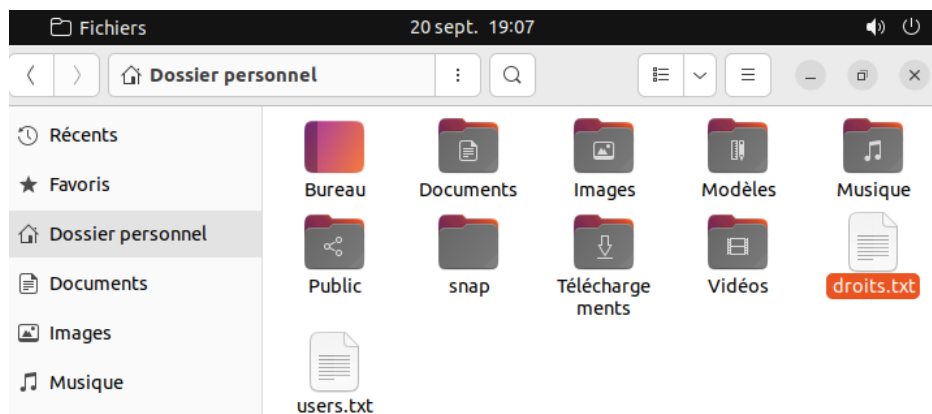
```
abaou-virtual-machine:~$ sudo usermod -a -G Plateformeurs User2
abaou-virtual-machine:~$
```

- Copier votre “users.txt” dans un fichier “droits.txt”

Tout d’abord on crée le document texte droits.txt avec la commande “touch”

```
abaou-virtual-machine:~$ touch droits.txt
```

Il est bien présent dans le bon répertoire :



- Copier votre “users.txt” dans un fichier “groupes.txt”

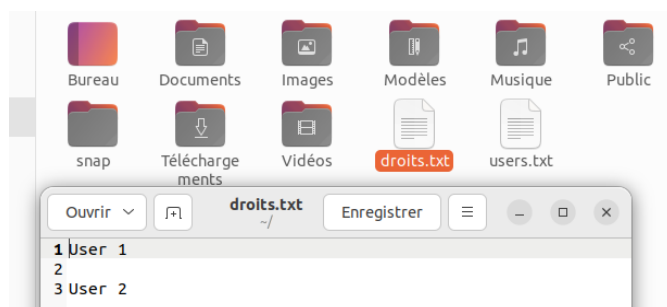
Pour copier le contenu d’un fichier vers un autre on fait :
 “cp nomdufichier1 nomdufichier2”.

nomdufichier1 étant le fichier dont on va copier le contenu, et
 nomdufichier2 étant le fichier où le contenu sera collé.

Ce qui donne : “cp users.txt droits.txt”

```
rabaou@rabaou-virtual-machine:~$ cp users.txt droits.txt
rabaou@rabaou-virtual-machine:~$
```

Et on voit que ça marche :



- Changer le propriétaire du fichier "droits.txt" pour mettre "User1"

C'est la commande chown qui permet de changer le propriétaire d'un fichier.

Voici la commande qu'on tape :

```
rabaou@rabaou-virtual-machine:~$ sudo chown User1 droits.txt
rabaou@rabaou-virtual-machine:~$
```

Pour vérifier si le changement a eu lieu, on tape cette commande dans la capture ci-dessous qui nous confirme qui est le nouveau propriétaire de notre fichier !

```
rabaou@rabaou-virtual-machine:~$ ls -l droits.txt
-rw-rw-r-- 1 User1 rabaou 15 sept. 20 19:06 droits.txt
rabaou@rabaou-virtual-machine:~$
```

Si on se souvient de la commande ls, ainsi que de la possibilité d'utiliser des options en plus, dont "-l" qui permet d'afficher les détails d'un fichier choisi à savoir : la date de création, le groupe, les autorisations liées au fichier et aussi le propriétaire.

(Revoir la question : Afficher les fichiers cachés plus les informations sur les droits sous forme de liste du JOB 01).

Remarque : En enlevant sudo comme ici on obtient un message qui nous dit que ce n'est pas permis :

```
rabaou@rabaou-virtual-machine:~$ chown User1 droits.txt
chown: modification du propriétaire de 'droits.txt': Opération non permise
```


Changer les droits du fichier "droits.txt" pour que "User2" ait accès seulement en lecture

Pour changer les droits on utilise la commande `chmod` avec les opérateurs `+` et `-` qui servent à donner ou enlever le droit de lire (`r`), exécuter (`x`) ou écrire (`w`) sur un fichier à ceux à qui s'applique le changement : `o` (autres), `a` (tout le monde), `u` (utilisateur propriétaire), `g` (groupe propriétaire).

Exemple : `chmod u+r texte.txt`

Cette commande donne le droit à l'utilisateur de pouvoir lire le fichier `texte.txt`

Cette commande `chmod` comporte néanmoins un problème. En effet je peux modifier le droit d'accès à un utilisateur mais l'utilisateur en question est le propriétaire actuel. En d'autres termes, je ne peux pas - ou alors je ne sais pas encore - faire une commande où je spécifie l'utilisateur dont je souhaite modifier les droits. Et même si j'essaie de le faire, j'obtiens un message d'erreur :

```
rabaou@rabaou-virtual-machine:~$ chmod User2+r droits.txt
chmod: mode incorrect : «User2+r»
Saisissez « chmod --help » pour plus d'informations.
```

Donc pour contourner ce problème, on va changer d'utilisateur à partir du terminal avec la commande `su` pour ensuite taper la commande suivante :

`chmod u+r droits.txt`

```
rabaou@rabaou-virtual-machine:~$ sudo su User2
$ chmod u+r droits.txt
chmod: impossible d'accéder à 'droits.txt': Permission non accordée
```

Je me retrouve avec ce message d'erreur. J'ai ensuite essayé de donner le droit d'accès à tous les utilisateurs pour que ce soit aussi le cas pour User2, mais j'ai eu le même résultat.

```
rabaou@rabaou-virtual-machine:~$ sudo chmod a+rxw droits.txt
[sudo] Mot de passe de rabaou :
rabaou@rabaou-virtual-machine:~$ sudo su User2
$ chmod u+r droits.txt
chmod: impossible d'accéder à 'droits.txt': Permission non accordée
```

Et lorsque j'essaie de forcer en mettant sudo avant la commande chmod, on me demande un mot de passe pour User2 alors que je n'en ai pas créé.

```
$ sudo chmod u+r droits.txt
[sudo] Mot de passe de User2 :
Désolé, essayez de nouveau.
[sudo] Mot de passe de User2 : █
```

Cela dit, il existe une autre solution qui se fait en trois étapes :

- Premièrement, je me connecte en tant que User1 dans le terminal (commande tapée : "su User1").
- Je vérifie si User2 appartient au groupe Plateformeurs (ce qui devrait être le cas).
(commande tapée : "groups User2")
- Si tel est le cas, je donne le droit de lire le fichier au groupe Plateformeurs. Ainsi, à fortiori, User2 se verra attribuer le droit de lire le fichier droits.txt.
(commande tapée : "chmod g+r droits.txt")

Changer les droits du fichier "groupes.txt" pour que les utilisateurs ait accès au fichier en lecture uniquement

On va utiliser de nouveau la commande chmod que l'on va changer au niveau des options :

"chmod a+r groupes.txt"

```
rabaou@rabaou-virtual-machine:~$ chmod a+r groupes.txt  
rabaou@rabaou-virtual-machine:~$
```

On rappelle que a signifie "all" donc tout le monde.

Changer les droits du fichier "groupes.txt" pour que les utilisateurs ait accès au fichier en lecture uniquement

On va utiliser de nouveau la commande chmod que l'on va changer au niveau des options :

"chmod g+rw groupes.txt"

```
rabaou@rabaou-virtual-machine:~$ chmod g+rw groupes.txt  
rabaou@rabaou-virtual-machine:~$
```

JOB 05

Premièrement, qu'est-ce qu'un alias ?

Un alias sert à créer une sorte d'alternative à des commandes qui seraient trop longues à taper. On peut définir un alias dans deux fichiers cachés du Dossier personnel qui sont :

- le fichier .bashrc
- un fichier .bash_aliases

Un alias sera aura toujours cette forme :
alias nom_alias = 'commande_alias'

- Ajouter un alias qui permettra de lancer la commande "ls -la" en tapant "la"

Pour ajouter cet alias on tape cette commande : alias la = 'ls -la'

```
rabaou@rabaou-virtual-machine:~$ alias ls='ls -la'
rabaou@rabaou-virtual-machine:~$ ls
total 100
drwxr-x--- 18 rabaou rabaou 4096 sept. 20 19:05 .
drwxr-xr-x  3 root   root   4096 sept. 19 18:38 ..
-rw-r----- 1 rabaou rabaou  57 sept. 20 20:46 .bash_history
-rw-r----- 1 rabaou rabaou 220 sept. 19 18:38 .bash_logout
-rw-r----- 1 rabaou rabaou 3771 sept. 19 18:38 .bashrc
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Bureau
drwx----- 15 rabaou rabaou 4096 sept. 20 19:07 .cache
drwx----- 13 rabaou rabaou 4096 sept. 20 19:07 .config
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Documents
-rwxrwxrwx  1 User1  rabaou  15 sept. 20 19:06 droits.txt
drwx-----  2 rabaou rabaou 4096 sept. 20 17:08 .gnupg
drwxr-xr-x  3 rabaou rabaou 4096 sept. 19 19:00 Images
-rw-r----- 1 rabaou rabaou  20 sept. 19 19:11 .lessht
drwx-----  3 rabaou rabaou 4096 sept. 19 18:55 .local
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Modèles
drwx-----  3 rabaou rabaou 4096 sept. 19 19:01 .mozilla
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Musique
-rw-r----- 1 rabaou rabaou  807 sept. 19 18:38 .profile
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Public
drwx-----  4 rabaou rabaou 4096 sept. 19 19:03 snap
drwx-----  2 rabaou rabaou 4096 sept. 19 18:57 .ssh
-rw-r----- 1 rabaou rabaou   0 sept. 19 23:41 .sudo_as_admin_successful
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Téléchargements
drwx-----  6 rabaou rabaou 4096 sept. 19 19:01 .thunderbird
-rw-rw-r--  1 rabaou rabaou  15 sept. 20 17:27 users.txt
drwxr-xr-x  2 rabaou rabaou 4096 sept. 19 18:55 Vidéos
```

- Ajouter un alias qui permettra de lancer la commande “apt-get update” en tapant “update”

alias update = 'apt-get update'

```
rabaou@rabaou-virtual-machine:~$ alias update='apt-get update'
rabaou@rabaou-virtual-machine:~$
```

- Ajouter un alias qui permettra de lancer la commande “apt-get upgrade” en tapant “upgrade”

On s’inspire de la commande que l’on a utilisé dans les deux premières réponses : alias upgrade = 'apt-get upgrade'

```
rabaou@rabaou-virtual-machine:~$ sudo apt-get upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
rabaou@rabaou-virtual-machine:~$
```

- Ajouter une variable d’environnement qui se nommera “USER” et qui sera égale à votre nom d’utilisateur

Une variable d’environnement est un moyen d’influencer le comportement d’un logiciel. A titre d’exemple, la variable d’environnement “LANG” détermine la langue utilisée pour communiquer avec l’utilisateur.

Pour créer une variable d’environnement, on tape la commande “export”.

Donc pour créer une variable d’environnement qui correspond au nom d’utilisateur, on fait :

“export USER=rabaou”

```
rabaou@rabaou-virtual-machine:~$ export USER=rabaou
rabaou@rabaou-virtual-machine:~$
```

- Afficher les variables d'environnement

La commande qui sert à afficher les variables d'environnement est "env".

```
rabaou@rabaou-virtual-machine:~$ env
SHELL=/bin/bash
SESSION_MANAGER=local/rabaou-virtual-machine:@/tmp/.ICE-unix/1075,unix/rabaou-virtual-machine:/tmp/.ICE-unix/1075
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
SSH_AGENT_LAUNCHER=gnome-keyring
no_proxy=localhost,127.0.0.0/8,::1
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
GNOME_SHELL_SESSION_MODE=ubuntu
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
GTK_MODULES=gail:atk-bridge
PWD=/home/rabaou
LOGNAME=rabaou
XDG_SESSION_DESKTOP=ubuntu
XDG_SESSION_TYPE=wayland
SYSTEMD_EXEC_PID=1108
XAUTHORITY=/run/user/1000/.mutter-Xwaylandauth.12KTS1
HOME=/home/rabaou
USERNAME=rabaou
IM_CONFIG_PHASE=1
LANG=fr_FR.UTF-8
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd=40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.lha=
```

- Mettre à jour les modifications de votre bashrc dans votre shell actuel

Pour actualiser le shell bash, il suffit d'utiliser la commande "exec bash" dans ~/.bash_aliases. Ainsi, on remplacera notre image shell par la plus récente, ce qui nous permettra d'utiliser les alias mis à jour.

```
rabaou@rabaou-virtual-machine:~$ exec bash
rabaou@rabaou-virtual-machine:~$
```

- Ajouter à votre Path le chemin "/home/'votre utilisateur'/Bureau"

Pour ajouter un répertoire au Path sur Linux , il suffit d'utiliser la commande export : export Path=\$PATH:/home/rabaou/Bureau

```
rabaou@rabaou-virtual-machine:~$ export Path=$PATH:/home/rabaou/Bureau
rabaou@rabaou-virtual-machine:~$
```

JOB 06

On a vu dans le JOB 03 que pour télécharger un fichier dont on a l'adresse on utilise la commande "wget" suivie de l'adresse.

Maintenant, pour extraire le contenu d'un fichier on utilise la commande : "tar -xzf" suivie du nom_du_fichier.

Donc les deux commandes que l'on tape sont :

"wget
<https://drive.google.com/file/d/11dSeIXQuH4tih6zesbv-6OMEpr-sT77X/view?usp=sharing>"

et "tar -xzf Shell.tar.gz"

JOB 07

- Créer un fichier “une_commande.txt” avec le texte suivant “Je suis votre fichier texte”

On peut créer le nouveau fichier avec la commande “touch”.
En revanche si on veut en plus de créer le fichier texte, écrire directement le contenu du document texte à partir du terminal, il faudrait utiliser la commande cat qui fonctionnerait comme ceci :

```
rabaou@rabaou-virtual-machine:~$ cat > une_commande.txt  
"Je suis votre fichier texte" rabaou@rabaou-virtual-machine:~$
```

“cat > une_commande.txt”

“Taper texte”, puis appuyer sur Entrée pour aller à la ligne (ce que je ne fais pas sur la capture d’écran).

Puis appuyer sur Ctrl+D pour enregistrer.

- Compter le nombre de lignes présentes dans votre fichier de source apt et les enregistrer dans un fichier nommé “nb_lignes.txt”

Pour compter le nombre de lignes d’un fichier on utilise la commande “wc” suivie de l’option “-l”.

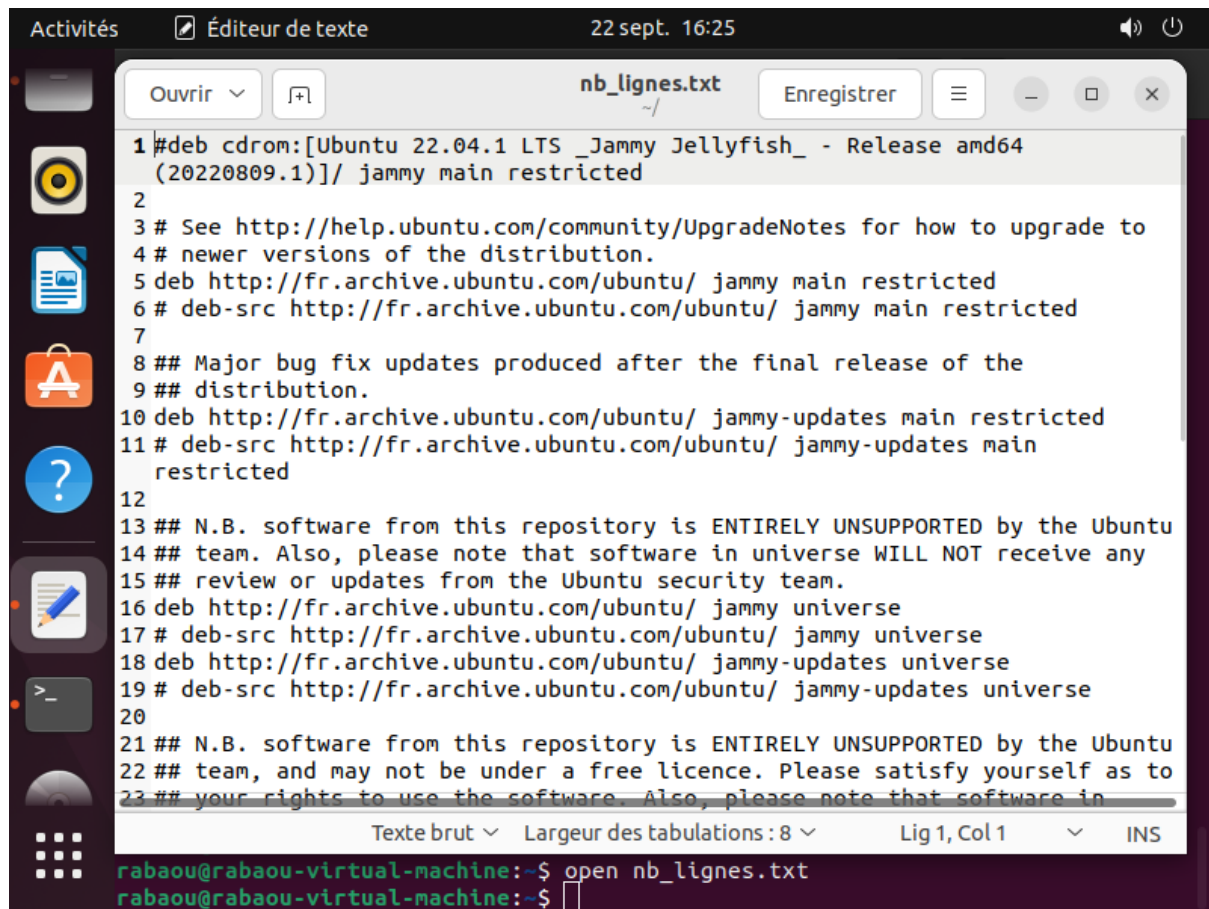
Ensuite on copie le tout avec cp dans un nouveau fichier créé avec touch.

On voit que l’on va réaliser plusieurs commandes en même temps donc on utilise les symboles “&&” pour toutes les exécuter en même temps.

Ce qui donne ceci :

```
rabaou@rabaou-virtual-machine:~$ wc -l /etc/apt/sources.list && touch nb_lignes  
.txt && cp /etc/apt/sources.list nb_lignes.txt  
50 /etc/apt/sources.list
```

On vérifie ensuite si tout a été bien réalisé en ouvrant nb_lignes.txt :



```
1 #deb cdrom:[Ubuntu 22.04.1 LTS _Jammy Jellyfish_ - Release amd64
  (20220809.1)]/ jammy main restricted
2
3 # See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
4 # newer versions of the distribution.
5 deb http://fr.archive.ubuntu.com/ubuntu/ jammy main restricted
6 # deb-src http://fr.archive.ubuntu.com/ubuntu/ jammy main restricted
7
8 ## Major bug fix updates produced after the final release of the
9 ## distribution.
10 deb http://fr.archive.ubuntu.com/ubuntu/ jammy-updates main restricted
11 # deb-src http://fr.archive.ubuntu.com/ubuntu/ jammy-updates main
   restricted
12
13 ## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
14 ## team. Also, please note that software in universe WILL NOT receive any
15 ## review or updates from the Ubuntu security team.
16 deb http://fr.archive.ubuntu.com/ubuntu/ jammy universe
17 # deb-src http://fr.archive.ubuntu.com/ubuntu/ jammy universe
18 deb http://fr.archive.ubuntu.com/ubuntu/ jammy-updates universe
19 # deb-src http://fr.archive.ubuntu.com/ubuntu/ jammy-updates universe
20
21 ## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
22 ## team, and may not be under a free licence. Please satisfy yourself as to
23 ## your rights to use the software. Also, please note that software in
```

Texte brut Largeur des tabulations : 8 Lig 1, Col 1 INS

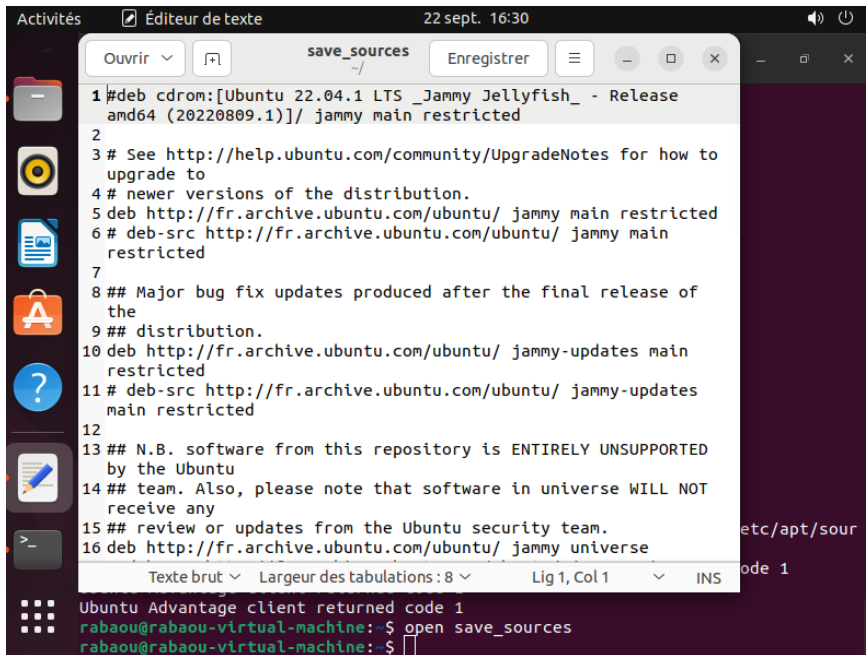
```
rabaou@rabaou-virtual-machine:~$ open nb_lignes.txt
rabaou@rabaou-virtual-machine:~$
```

- Afficher le contenu du fichier source apt et l'enregistrer dans un autre fichier appelé "save_sources"

Pour afficher le contenu du fichier on utilise la commande open et pour enregistrer dans save_sources on utilise la commande cp.

```
rabaou@rabaou-virtual-machine:~$ open /etc/apt/sources.list && cp /etc/apt/sources.list save_sources
rabaou@rabaou-virtual-machine:~$ Ubuntu Advantage client returned code 1
Ubuntu Advantage client returned code 1
Ubuntu Advantage client returned code 1
rabaou@rabaou-virtual-machine:~$
```

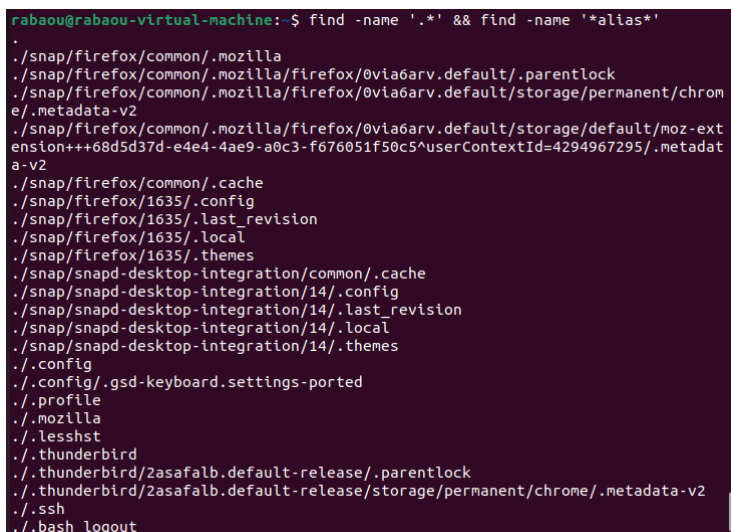
On peut ensuite vérifier le fichier save_sources :



```
1 #deb cdrom:[Ubuntu 22.04.1 LTS _Jammy Jellyfish_ - Release  
amd64 (20220809.1)]/ jammy main restricted  
2  
3 # See http://help.ubuntu.com/community/UpgradeNotes for how to  
upgrade to  
4 # newer versions of the distribution.  
5 deb http://fr.archive.ubuntu.com/ubuntu/ jammy main restricted  
6 # deb-src http://fr.archive.ubuntu.com/ubuntu/ jammy main  
restricted  
7  
8 ## Major bug fix updates produced after the final release of  
the  
9 ## distribution.  
10 deb http://fr.archive.ubuntu.com/ubuntu/ jammy-updates main  
restricted  
11 # deb-src http://fr.archive.ubuntu.com/ubuntu/ jammy-updates  
main restricted  
12  
13 ## N.B. software from this repository is ENTIRELY UNSUPPORTED  
by the Ubuntu  
14 ## team. Also, please note that software in universe WILL NOT  
receive any  
15 ## review or updates from the Ubuntu security team.  
16 deb http://fr.archive.ubuntu.com/ubuntu/ jammy universe  
etc/apt/sour  
code 1  
rabaou@rabaou-virtual-machine:~$ open save_sources  
rabaou@rabaou-virtual-machine:~$
```

- Faites une recherche des fichiers commençant par “.” tout en cherchant le mot alias qui sera utilisé depuis un fichier

Pour faire une recherche avancée, on utilise la commande find. Pour compléter afin de donner une indication de ce que l’on veut chercher on ajoute “ -name ‘.*’ ” (comme sur cet exemple)



```
rabaou@rabaou-virtual-machine:~$ find -name '.*' && find -name '*alias*'  
./snap/firefox/common/.mozilla  
./snap/firefox/common/.mozilla/firefox/0via6arv.default/.parentlock  
./snap/firefox/common/.mozilla/firefox/0via6arv.default/storage/permanent/chrome/.metadata-v2  
./snap/firefox/common/.mozilla/firefox/0via6arv.default/storage/default/moz-extension+++68d5d37d-e4e4-4ae9-a0c3-f676051f50c5^userContextId=4294967295/.metadata-v2  
./snap/firefox/common/.cache  
./snap/firefox/1635/.config  
./snap/firefox/1635/.last_revision  
./snap/firefox/1635/.local  
./snap/firefox/1635/.themes  
./snap/snapd-desktop-integration/common/.cache  
./snap/snapd-desktop-integration/14/.config  
./snap/snapd-desktop-integration/14/.last_revision  
./snap/snapd-desktop-integration/14/.local  
./snap/snapd-desktop-integration/14/.themes  
./config  
./config/.gsd-keyboard.settings-ported  
./profile  
./mozilla  
./lessshst  
./thunderbird  
./thunderbird/2asafalb.default-release/.parentlock  
./thunderbird/2asafalb.default-release/storage/permanent/chrome/.metadata-v2  
./ssh  
./bash_logout
```

On fait donc une commande où on ajoute avec “&&” la commande qui sert à chercher le mot alias : la commande correspondante est la commande grep (on fera grep -r alias).

Je n’ai pas pu afficher tout le résultat car il est trop long mais voilà en gros ce qu’on a dans ces captures d’écran :

```
rabaou@rabaou-virtual-machine:~$ find -name '.*' && grep -r alias
./snap/firefox/common/.mozilla
./snap/firefox/common/.mozilla/firefox/0via6arv.default/.parentlock
./snap/firefox/common/.mozilla/firefox/0via6arv.default/storage/permanent/chrome/.metadata-v2
./snap/firefox/common/.mozilla/firefox/0via6arv.default/storage/default/moz-extension+++68d5d37d-e4e4-4ae9-a0c3-f676051f50c5^userContextId=4294967295/.metadata-v2
./snap/firefox/common/.cache
./snap/firefox/1635/.config
./snap/firefox/1635/.last_revision
./snap/firefox/1635/.local
./snap/firefox/1635/.themes
./snap/snapd-desktop-integration/common/.cache
./snap/snapd-desktop-integration/14/.config
./snap/snapd-desktop-integration/14/.last_revision
./snap/snapd-desktop-integration/14/.local
./snap/snapd-desktop-integration/14/.themes
./config
./config/.gsd-keyboard.settings-ported
./profile
./mozilla
./lessht
./thunderbird
./thunderbird/2asafalb.default-release/.parentlock
./thunderbird/2asafalb.default-release/storage/permanent/chrome/.metadata-v2
./ssh
./bash_logout
```

```
rabaou@rabaou-virtual-machine: ~
grep: .thunderbird/2asafalb.default-release/blist.sqlite : fichiers binaires co
rrespondent
.bashrc:# enable color support of ls and also add handy aliases
.bashrc:  alias ls='ls --color=auto'
.bashrc:  #alias dir='dir --color=auto'
.bashrc:  #alias vdir='vdir --color=auto'
.bashrc:  alias grep='grep --color=auto'
.bashrc:  alias fgrep='fgrep --color=auto'
.bashrc:  alias egrep='egrep --color=auto'
.bashrc:# some more ls aliases
.bashrc:alias ll='ls -lF'
.bashrc:alias la='ls -lA'
.bashrc:alias l='ls -CF'
.bashrc:# Add an "alert" alias for long running commands. Use like so:
.bashrc:alias alert='notify-send --urgency=low -i "[ $? = 0 ] && echo termina
l || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[\;&]\s*a
lert$//'\''")'
.bashrc:# ~/.bash_aliases, instead of adding them here directly.
.bashrc:if [ -f ~/.bash_aliases ]; then
.bashrc:  . ~/.bash_aliases
.bash_history:alias update = "apt-get update"
.bash_history:alias update = 'apt-get update'
.bash_history:alias update='apt-get update'
.bash_history:alias ls='ls -la'
grep: .cache/thunderbird/2asafalb.default-release/startupCache/startupCache.8.l
ittle : fichiers binaires correspondant
grep: .cache/thunderbird/2asafalb.default-release/startupCache/webext.sc.lz4 :
fichiers binaires correspondant
rabaou@rabaou-virtual-machine:~$
```

POUR ALLER PLUS LOIN

Comme je l'ai expliqué pour une partie du JOB 03, je n'ai pas de connexion internet sur Ubuntu donc je n'ai pas pu installé de commande "tree" pour réaliser cette partie.