



Projet “Shell”

Mohammed Yassine Dabboussi

Sommaire

Introduction	3
Job 1	4
Job 2	6
Job 3	6
Job 4	8
Job 5	10
Job 6	12
Job 7	13
Pour aller plus loin	15

Introduction

Système d'exploitation

En informatique, un système d'exploitation est un ensemble de programmes qui dirige l'utilisation des ressources d'un ordinateur par des logiciels applicatifs.

Linux ou GNU/Linux est une famille de systèmes d'exploitation open source de type Unix fondé sur le noyau Linux, créé en 1991 par Linus Torvalds. De nombreuses distributions Linux ont depuis vu le jour et constituent un important vecteur de popularisation du mouvement du logiciel libre.

Si à l'origine Linux a été développé pour les ordinateurs compatibles PC, il n'a jamais équipé qu'une très faible part des ordinateurs personnels. Mais le noyau Linux, accompagné ou non des logiciels GNU, est également utilisé par d'autres types de systèmes informatiques, notamment les serveurs, téléphones portables, systèmes embarqués ou encore superordinateurs. Le système d'exploitation pour téléphones portables Android qui utilise le noyau Linux mais pas GNU, équipe aujourd'hui 85 % des tablettes tactiles et smartphones.

Kernel

Kernel est un terme anglais qui signifie en français « noyau ». Il constitue le cœur du système qui exploite un ordinateur. Il établit la communication entre la partie matérielle et la partie logicielle de l'appareil. C'est ce qui permet à l'utilisateur d'interagir avec la machine.

Terminal

(116 101 114 109 105 110 97 108 32 108 105 110 117 120 == terminal linux)

Un terminal, ou interface en ligne de commande, est une interface homme-machine dans laquelle l'utilisateur interagit avec la machine en mode texte. L'utilisateur écrit des lignes de commande, la machine les exécute et affiche le résultat des commandes.

Shell est un programme qui sert d'interface en mode texte entre le noyau et l'utilisateur, il est aussi un interpréteur de commande et un langage de programmation.

Le shell est ainsi chargé de faire l'intermédiaire le système d'exploitation et l'utilisateur grâce aux lignes de commandes saisies par ce dernier. Son rôle consiste ainsi à lire la ligne de commande, interpréter sa signification, exécuter la commande, puis retourner le résultat sur les sorties.

Job 1

man ls

La commande `man` permet d'afficher le manuel d'une commande. Par exemple `man ls` permet d'afficher les options de la commande `ls`.

La commande `ls` affiche le contenu d'un répertoire.



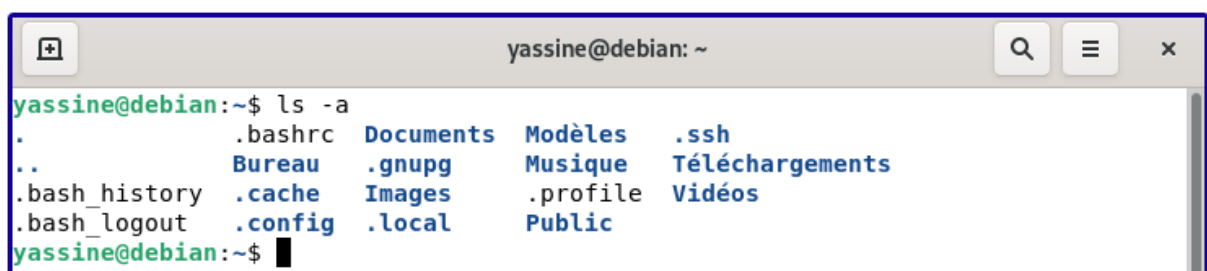
```
yassine@debian: ~  
yassine@debian:~$ ls  
Bureau Documents Images Modèles Musique Public Téléchargements Vidéos  
yassine@debian:~$
```

Dossiers et fichiers cachés

Les éléments cachés sont des éléments qui ne sont pas visibles par défaut lorsque l'utilisateur affiche le contenu d'un dossier. Cette possibilité de dissimuler des éléments permet de protéger des fichiers de manipulations involontaires et d'alléger l'affichage des dossiers dans lesquels ils sont stockés.

Par défaut, la commande `ls` n'affiche pas les fichiers commençant par un point '.' (c'est-à-dire les fichiers cachés).

L'option `-a` affiche les fichiers cachés (ceux commençant par un point).



```
yassine@debian: ~  
yassine@debian:~$ ls -a  
.          .bashrc  Documents Modèles   .ssh  
..         Bureau  .gnupg   Musique  Téléchargements  
.bash_history .cache  Images   .profile  Vidéos  
.bash_logout .config .local   Public  
yassine@debian:~$
```

L'option `-l` de la commande `ls` permet d'afficher les informations suivantes :

- le type de fichier
 - `-` pour les fichiers ordinaires
 - `d` pour les dossiers
 - `b` pour les fichiers spéciaux en mode bloc
 - `c` pour les fichiers spéciaux en mode caractère
 - `l` pour les liens symboliques
 - `p` pour les fifo (pipes)

- **s** pour les socket
- les permissions d'accès
- le nombre de liens physiques
- le nom du propriétaire et du groupe
- la taille en octets
- l'horodatage (date de dernière modification)

En associant les deux options **-a** et **-l** en même temps à la commande **ls** ça permet d'afficher les fichiers cachés plus les informations sur les droits sous forme de liste

```

yassine@debian: ~
yassine@debian:~$ ls -a -l
total 76
drwxr-xr-x 15 yassine yassine 4096 20 sept. 10:55 .
drwxr-xr-x  3 root    root    4096 19 sept. 16:56 ..
-rw-----  1 yassine yassine   62 20 sept. 10:53 .bash_history
-rw-r--r--  1 yassine yassine  220 19 sept. 16:56 .bash_logout
-rw-r--r--  1 yassine yassine 3526 19 sept. 16:56 .bashrc
drwxr-xr-x  2 yassine yassine 4096 19 sept. 16:59 Bureau
drwx----- 10 yassine yassine 4096 19 sept. 22:09 .cache
drwx----- 10 yassine yassine 4096 19 sept. 21:23 .config
drwxr-xr-x  2 yassine yassine 4096 19 sept. 16:59 Documents
drwx-----  2 yassine yassine 4096 20 sept. 10:54 .gnupg
drwxr-xr-x  2 yassine yassine 4096 19 sept. 16:59 Images
drwxr-xr-x  3 yassine yassine 4096 19 sept. 16:59 .local
drwxr-xr-x  2 yassine yassine 4096 19 sept. 16:59 Modèles
drwxr-xr-x  2 yassine yassine 4096 19 sept. 16:59 Musique
-rw-r--r--  1 yassine yassine   807 19 sept. 16:56 .profile
drwxr-xr-x  2 yassine yassine 4096 19 sept. 16:59 Public
drwx-----  2 yassine yassine 4096 19 sept. 21:23 .ssh
drwxr-xr-x  2 yassine yassine 4096 19 sept. 16:59 Téléchargements
drwxr-xr-x  2 yassine yassine 4096 19 sept. 16:59 Vidéos
yassine@debian:~$

```

Comment ajouter des options à une commande ?

Le 1er mot tapé est une commande. Les mots tapés après un tiret sont des options courtes et les mots tapés après 2 tirets sont des options longues, le reste ce sont des paramètres.

Notez qu'il existe des commandes sans paramètres, d'autres sans options, et même certaines qui prennent une commande en paramètres .

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

commande -option

commande --option

Job 2

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

cat elle sert le plus souvent à lire un contenu textuel (imprimer des fichiers dans l'écran), mais elle peut faire plus.

Exemple: **cat** .bashrc

- afficher les 10 premières lignes du fichier “.bashrc”

La commande **head** suivie d'un nom de fichier, sans préciser d'options affiche les 10 premières lignes du fichier.

Pour utiliser cette commande, il n'est pas utile d'avoir un accès root sur la machine, à moins qu'on cherche à lire le contenu d'un fichier protégé auquel cas il sera nécessaire d'avoir suffisamment de privilèges.

- afficher les 10 dernières lignes du fichier “.bashrc”

La commande **tail** permet de visionner les dernières lignes d'un fichier texte. Par défaut celui-ci affiche que les dix dernières lignes.

tail .bashrc

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

head -20 .bashrc || **head** -n 20 .bashrc

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

tail -20 .bashrc || **tail** -n 20 .bashrc

Job 3

Installer le paquet “cmatrix”

- Il faut passer d'abord en mode super-utilisateur en tapant cette commande :

su -

su bascule sur le compte d'utilisateur root et requiert le mot de passe du compte root.

Après il faut taper la commande qui permet d'installer le paquet cmatrix :

apt install cmatrix

- ```
apt -get upgrade
```

La commande **upgrade** permet d'installer les mises à jour disponibles de tous les paquets présents sur le système.

## Télécharger les internet : Google

1) Télécharger la page google :

**wget** <https://www.google.com>

2) Télécharger le navigateur Google chrome :

2.1) Télécharger la version 64bits de Google Chrome au format DEB avec cette commande :

**wget** [https://dl.google.com/linux/direct/google-chrome-stable\\_current\\_amd64.deb](https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb)

2.2) Installer le paquet téléchargé :

**sudo apt install** ./google-chrome-stable\_current\_amd64.deb

ou **su**

**apt install** ./google-chrome-stable\_current\_amd64.deb

## Redémarrer votre machine

**systemctl reboot** ou **sudo reboot**

## éteindre votre machine

**systemctl poweroff** ou **sudo poweroff**

**systemctl halt** (mot de passe nécessaire pour cette commande)

**shutdown** : pour arrêter le système proprement et en alertant les utilisateurs connectés (si plusieurs sessions ouvertes)

**halt, poweroff et reboot** : pour arrêter le système en tuant tous les processus en cours et sans avertir personne.

## Job 4

Créer un fichier **users.txt** qui contiendra **User1** et **User2** séparé par un retour à la ligne

**echo -e "user1\nuser2" >> users.txt**

**echo** : Commande permettant de créer le fichier texte en lui



-e : option permet l'interprétation des échappements antislash

"user1\nuser2" : le texte à ajouter dans le fichier

\n : retour à la ligne

>> : représente la redirection de sortie

users.txt : le fichier texte à créer et qui contiendra le text ci-dessus

### **Créer un groupe appelé "Plateformeurs"**

Les groupes servent à rassembler des utilisateurs afin de leur attribuer des droits communs.

**su -**

**addgroup** plateformeurs

### **Créer un utilisateur appelé "User1"**

**sudo adduser user1** ou **sudo useradd user1**

### **Créer un utilisateur appelé "User2"**

**sudo adduser user2** ou **sudo useradd user2**

### **Ajouter "User2" au groupe Plateformeurs**

**usermod -a -G** plateformeurs user2

### **Copier votre "users.txt" dans un fichier "droits.txt"**

**cp** [OPTIONS] SOURCE... DESTINATION

**cp** users.txt droits.txt

ou **cp -r** si on va copier dans deux fichier ou plus

### **Copier votre "users.txt" dans un fichier "groupes.txt"**

**cp** users.txt groupes.txt

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

**sudo chown** user1 droits.txt

### **Changer les droits du fichier "droits.txt" pour que "User2" ai accès seulement en lecture**

**sudo chmod permissions fichier**

**sudo chmod 644 droits.txt**

**644: u(r+w), g(r), o(r)**

**ls -l droits.txt**

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

**sudo chmod 444 groupes.txt**

Changer les droits du fichier pour que le groupe "Plateformeurs" puissent y accéder en lecture/écriture.

**sudo chgrp plateformeurs groupes.txt**

## Job 5

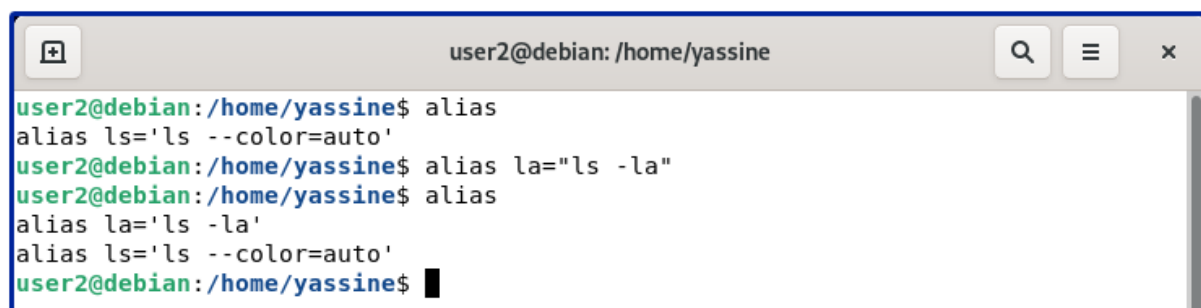
### Alias

Les alias permettent de définir des raccourcis pour les commandes.

Quand on tape la commande alias, on a comme résultat la liste des alias présents.

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

**alias la="ls -la"**

A terminal window titled 'user2@debian: /home/yassine' with search, menu, and close buttons. The terminal shows the following commands and output:

```
user2@debian:/home/yassine$ alias
alias ls='ls --color=auto'
user2@debian:/home/yassine$ alias la="ls -la"
user2@debian:/home/yassine$ alias
alias la='ls -la'
alias ls='ls --color=auto'
user2@debian:/home/yassine$
```

**Ps :** L'alias créé ci-dessus est temporaire, si on veut le garder pour toujours il faut écrire aussi la commande sur le fichier bashrc.

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

**alias update="apt-get update"**

```
user2@debian: /home/yassine
user2@debian:/home/yassine$ alias
alias la='ls -la'
alias ls='ls --color=auto'
user2@debian:/home/yassine$ alias update="apt-get update"
user2@debian:/home/yassine$ alias
alias la='ls -la'
alias ls='ls --color=auto'
alias update='apt-get update'
user2@debian:/home/yassine$
```

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

**alias** upgrade="apt-get upgrade"

```
user2@debian: /home/yassine
user2@debian:/home/yassine$ alias
alias la='ls -la'
alias ls='ls --color=auto'
alias update='apt-get update'
user2@debian:/home/yassine$ alias upgrade="apt-get upgrade"
user2@debian:/home/yassine$ alias
alias la='ls -la'
alias ls='ls --color=auto'
alias update='apt-get update'
alias upgrade='apt-get upgrade'
user2@debian:/home/yassine$
```

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

Une **variable d’environnement** est une variable dynamique utilisée par des processus ou des applications afin de définir des chemins ou des raccourcis d’information.

La commande **env** permet d’afficher toutes les variables d’environnement présentes sur le système.

**export** USER=\$USERNAME et pour l’afficher: **echo** \$USER

```
user2@debian: /home/yassine
user2@debian:/home/yassine$ export USER=$USERNAME
user2@debian:/home/yassine$ echo $USER
yassine
user2@debian:/home/yassine$
```

(Pour supprimer la variable: **unset USER** )

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

ouvrir le fichier .bashrc et le modifier:

**nano .bashrc**

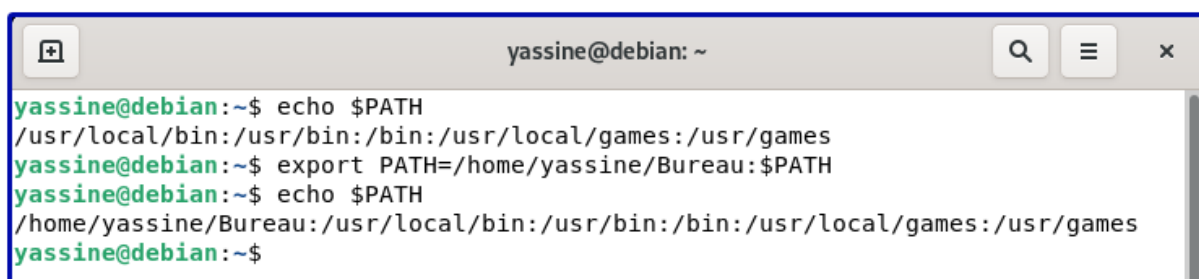
ajouter la commande dans le fichier.

### - Afficher les variables d'environnement

La commande **env** permet d'afficher toutes les variables d'environnement présentes sur le système.

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

Ce sont les répertoires dans lesquels le shell cherche la commande qu'on écrit au clavier. La recherche se fait dans l'ordre des répertoires contenus dans la variable PATH.

A terminal window titled 'yassine@debian: ~' showing the following commands and output:

```
yassine@debian:~$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
yassine@debian:~$ export PATH=/home/yassine/Bureau:$PATH
yassine@debian:~$ echo $PATH
/home/yassine/Bureau:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
yassine@debian:~$
```

## Job 6

### 1) Télécharger l'archive:

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

### 2) Désarchiver:

**tar -xf** Copie\ de\ Ghost\ in\ the\ shell.tar.gz

### 3) Ouvrir le fichier PDF:

**open** Ghost\ in\ the\ shell.pdf

**evince** Ghost\ in\ the\ shell.pdf

```
yassine@debian: ~/Téléchargements
yassine@debian: ~/Téléchargements
yassine@debian:~/Téléchargements$ ls
'Copie de Ghost in the Shell.tar.gz'
yassine@debian:~/Téléchargements$ man tar
yassine@debian:~/Téléchargements$ tar -xf Copie\ de\ Ghost\ in\ the\ Shell.tar.g
z
yassine@debian:~/Téléchargements$ ls
'Copie de Ghost in the Shell.tar.gz' 'Ghost in the Shell.pdf'
yassine@debian:~/Téléchargements$ evince Ghost\ in\ the\ Shell.pdf
yassine@debian:~/Téléchargements$
```

## Job 7

- Créer un fichier “une\_commande.txt” avec le texte suivant “Je suis votre fichier texte”

echo “Je suis votre fichier texte” >> une\_commande.txt

```
yassine@debian: ~
yassine@debian:~$ ls
Bureau droits.txt Images Musique Téléchargements Vidéos
Documents groupes.txt Modèles Public users.txt
yassine@debian:~$ echo "Je suis votre fichier texte" >> une_commande.txt
yassine@debian:~$ ls
Bureau droits.txt Images Musique Téléchargements users.txt
Documents groupes.txt Modèles Public une_commande.txt Vidéos
yassine@debian:~$ cat une_commande.txt
Je suis votre fichier texte
yassine@debian:~$
```

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

wc -l /etc/apt/sources.list > /home/yassine/nb\_lignes.txt

```
yassine@debian: ~
yassine@debian:~$ ls
Bureau groupes.txt Musique Téléchargements Vidéos
Documents Images Public une_commande.txt
droits.txt Modèles save_sources.txt users.txt
yassine@debian:~$ wc -l /etc/apt/sources.list > /home/yassine/nb_lignes.txt
yassine@debian:~$ cat nb_lignes.txt
20 /etc/apt/sources.list
yassine@debian:~$
```

Autres options wc :

wc -l: Affiche le nombre de lignes d'un fichier.

wc -w: Affiche le nombre de mots d'un fichier.

wc -c: Affiche le nombre d'octets dans un fichier.

wc -m: Affiche le nombre de caractères d'un fichier.

wc -L: Affiche uniquement la longueur de la plus longue ligne d'un fichier.

**- Afficher le contenu du fichier source apt et l'enregistrer dans un autre fichier appelé "save\_sources"**

**cat** /etc/apt/sources.list ; **cat** /etc/apt/sources.list > /home/yassine/save\_sources.txt

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

**find** est un outil de recherche disposant d'un grand nombre d'options. Cette commande peut également enchaîner des actions sur le résultat des recherches et utiliser des expressions régulières comme motifs de recherche.

**find -type f -name "." | grep -rnw "alias"**

```
.bashrc:78: alias ls='ls --color=auto'
.bashrc:79: #alias dir='dir --color=auto'
.bashrc:80: #alias vdir='vdir --color=auto'
.bashrc:82: #alias grep='grep --color=auto'
.bashrc:83: #alias fgrep='fgrep --color=auto'
.bashrc:84: #alias egrep='egrep --color=auto'
.bashrc:91:#alias ll='ls -l'
.bashrc:92:#alias la='ls -A'
.bashrc:93:#alias l='ls -CF'
```

**find** est un outil de recherche disposant d'un grand nombre d'options. Cette commande peut également enchaîner des actions sur le résultat des recherches et utiliser des expressions régulières comme motifs de recherche.

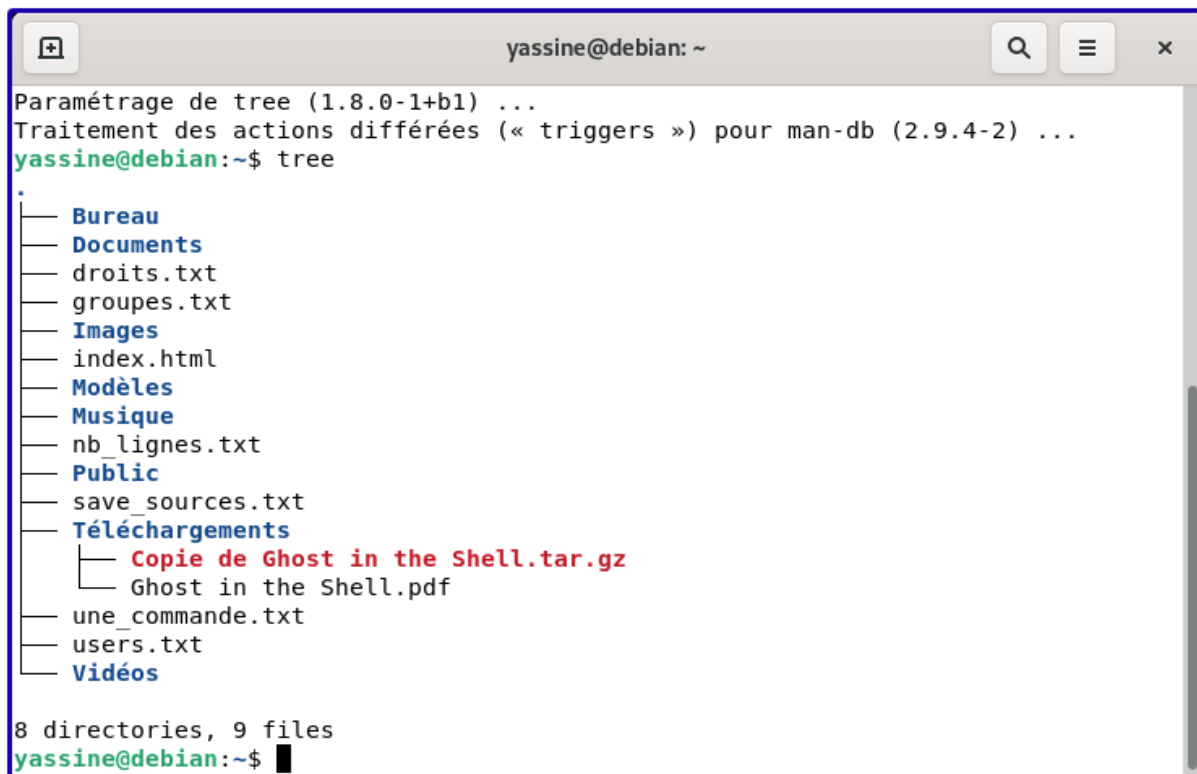
**grep** est une commande utilisé pour rechercher une chaîne de caractères dans un fichier spécifié. Lorsqu'il trouve une correspondance, il imprime la ligne avec le résultat.

La commande grep est pratique lors de la recherche dans de gros fichiers .

# Pour aller plus loin

-Installer la commande tree

**sudo apt install tree**

A terminal window titled 'yassine@debian: ~' showing the output of the 'tree' command. The output displays a directory tree structure starting from the current directory ('.'), listing various folders like 'Bureau', 'Documents', 'Images', 'Modèles', 'Musique', 'Public', 'Téléchargements', and 'Vidéos', along with files like 'droits.txt', 'groupes.txt', 'index.html', 'nb\_lignes.txt', 'save\_sources.txt', 'une\_commande.txt', 'users.txt', and 'Ghost in the Shell.pdf'. The 'Téléchargements' folder is expanded, showing its contents. At the bottom, it states '8 directories, 9 files' and the prompt 'yassine@debian:~\$' is visible.

```
Paramétrage de tree (1.8.0-1+b1) ...
Traitement des actions différées (« triggers ») pour man-db (2.9.4-2) ...
yassine@debian:~$ tree
.
├── Bureau
├── Documents
├── droits.txt
├── groupes.txt
├── Images
├── index.html
├── Modèles
├── Musique
├── nb_lignes.txt
├── Public
├── save_sources.txt
├── Téléchargements
│ ├── Copie de Ghost in the Shell.tar.gz
│ └── Ghost in the Shell.pdf
├── une_commande.txt
├── users.txt
└── Vidéos

8 directories, 9 files
yassine@debian:~$
```

- Lancer la commande tree en arrière-plan qui aura pour but d'afficher toute l'arborescence en de votre / en enregistrant le résultat dans un fichier "tree.save"

**tree / > tree.save**

- lister les éléments présents dans le dossier courant est utilisé directement le résultat de votre première commande pour compter le nombre d'éléments trouvés

**ls ; ls | wc -l**

- Lancer une commande pour update vos paquets, si l'update réussit alors, vous devrez lancer un upgrade de vos paquets. Si l'update échoue, votre upgrade ne se lancera pas

**sudo apt-get update && sudo apt-get upgrade**