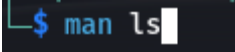


Job 1

Pour afficher le manuel de la commande **ls** nous devons mettre **man** avant, ça nous donne  et le manuel de la commande s'affiche;

```
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 specified.

  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
  --block-size=SIZE
      with -l, scale sizes by SIZE when printing them; e.g., '--block-size=M'; see SIZE format below
  -B, --ignore-backups
      do not list implied entries ending with ~
  -c
      with -lt: sort by, and show, ctime (time of last modification of file status information); with -l:
      show ctime and sort by name; otherwise: sort by ctime, newest first
  -C
      list entries by columns
  --color[=WHEN]
      color the output WHEN; more info below
  -d, --directory
      list directories themselves, not their contents
  -D, --dired
      generate output designed for Emacs' dired mode
  -f
      list all entries in directory order
  -F, --classify[=WHEN]
      Manual page ls(1) line 1 (press h for help or q to quit)
```

Pour afficher les fichier caché du home de notre utilisateur nous faisons **ls -a**,
-a défini all (tout) dans les commandes supplémentaire à **ls**.

```
rapatt@debian:~$ ls -a
.          .bashrc  Desktop    .face      .local     .profile   Templates
..         .cache   Documents  .face.icon  Music      Public     Videos
.bash_logout .config  Downloads  .gnupg     Pictures   .ssh
rapatt@debian:~$
```

Enfin pour afficher les fichiers cachés plus les informations des droits sous forme de listes, nous faisons ; **ls -la** dans notre terminal.

```
rapatt@debian:~$ ls -la
total 104
drwx----- 16 rapatt rapatt 4096 Sep 27 12:25 .
drwxr-xr-x  3 root   root   4096 Sep 27 10:38 ..
-rw-----  1 rapatt rapatt   66 Sep 27 10:46 .bash_history
-rw-r--r--  1 rapatt rapatt  220 Sep 27 10:38 .bash_logout
-rw-r--r--  1 rapatt rapatt 3526 Sep 27 10:38 .bashrc
drwx----- 13 rapatt rapatt 4096 Sep 27 11:21 .cache
drwx-----  9 rapatt rapatt 4096 Sep 27 11:17 .config
drwxr-xr-x  2 rapatt rapatt 4096 Sep 27 10:41 Desktop
drwxr-xr-x  2 rapatt rapatt 4096 Sep 27 10:41 Documents
drwxr-xr-x  2 rapatt rapatt 4096 Sep 27 11:21 Downloads
-rw-r--r--  1 rapatt rapatt 5290 Sep 27 10:38 .face
lrwxrwxrwx  1 rapatt rapatt    5 Sep 27 10:38 .face.icon -> .face
-rw-r--r--  1 rapatt rapatt  386 Sep 27 12:29 fichiers_alias.txt
drwx-----  2 rapatt rapatt 4096 Sep 27 10:46 .gnupg
drwx-----  4 rapatt rapatt 4096 Sep 27 10:41 .local
drwx-----  4 rapatt rapatt 4096 Sep 27 10:47 .mozilla
drwxr-xr-x  2 rapatt rapatt 4096 Sep 27 10:41 Music
-rw-r--r--  1 rapatt rapatt    3 Sep 27 12:29 nb_lignes.txt
drwxr-xr-x  2 rapatt rapatt 4096 Sep 27 10:41 Pictures
-rw-r--r--  1 rapatt rapatt  807 Sep 27 10:38 .profile
drwxr-xr-x  2 rapatt rapatt 4096 Sep 27 10:41 Public
-rw-r--r--  1 rapatt rapatt   63 Sep 27 12:29 save_sources
drwx-----  2 rapatt rapatt 4096 Sep 27 10:41 .ssh
drwxr-xr-x  2 rapatt rapatt 4096 Sep 27 10:41 Templates
-rw-r--r--  1 rapatt rapatt   28 Sep 27 12:29 une_commande.txt
drwxr-xr-x  2 rapatt rapatt 4096 Sep 27 10:41 Videos
rapatt@debian:~$
```

Question:

- Comment ajouter des options à une commande ?

Pour ajouter une option de commande nous mettons dans le terminal - suivi d'une lettre par exemple **ls -a**.

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

Dans les syntaxes nous retrouvons les lettres après le tiret pour les options et nous avons aussi les mots après 2 tirets. par exemple **man -h** et **man --help**

Job 2

Pour lire un fichier nous utilisons la commandes **cat** (*concatenate*) ce qui nous donne,

```
rapatt@debian:~/Documents$ ls
bout.txt
rapatt@debian:~/Documents$ cat bout.txt
c'est un fichier pour le job créer par la commande cat
```

Grâce à la commande **ls** nous voyons notre fichier en [.txt] puis nous l'ouvrons avec la fonction **cat**.

Pour afficher les 10 premières ligne du fichier .bashrc nous tapons la commande **head .bashrc** ce qui nous donne ;

```
rapatt@debian:~$ head .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

rapatt@debian:~$ █
```

Pour afficher les 10 dernières lignes du fichier .bashrc nous faisons **tail .bashrc** et nous avons;

```
rapatt@debian:~$ tail .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

rapatt@debian:~$ █
```

Ainsi pour avoir les 20 premières ou dernières ligne nous devons juste mettre **-20** entre **head/tail** et **.bashrc** et nous obtenons;

```
rapatt@debian:~$ head -20 .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
rapatt@debian:~$
```

```
rapatt@debian:~$ tail -20 .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
rapatt@debian:~$
```

Job 3

Pour installer le paquet cmatrix nous faisons dans notre terminal

```
root@debian:~# apt-get install cmatrix
```

Ici je l'ai fait en tant que root mais sinon en tant qu'utilisateur nous faisons **sudo apt-get install cmatrix**

De plus, pour lancer le paquet que nous venons d'installer (cmatrix) nous avons juste taper cMatrix dans le terminal et nous avons un effet matrix dans celui-ci.

Pour mettre à jour notre gestionnaire de paquet nous faisons **sudo apt-get upgrade** ou bien si on est déjà en tant qu'utilisateur root nous faisons **apt-get upgrade**.

```
root@debian:~# apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@debian:~#
```

Pour mettre à jour nos logiciels nous faisons **sudo apt-get update** ou **apt-get update** suivant si on est super utilisateur ou non.

```
root@debian:~# apt-get update
Hit:1 http://deb.debian.org/debian bookworm InRelease
Hit:2 http://security.debian.org/debian-security bookworm-security InRelease
Hit:3 http://deb.debian.org/debian bookworm-updates InRelease
Reading package lists... Done
root@debian:~#
```

En procédant à l'installation de l'internet Google nous avons utilisé la commande **wget** et incluent l'url de téléchargement de google nous pouvons donc l'installer depuis le terminal.

Ce qui nous donne alors ;

wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb

```
root@debian:~# wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
--2023-09-20 15:46:07-- https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
Resolving dl.google.com (dl.google.com)... 142.251.37.46, 2a00:1450:4006:80c::200e
Connecting to dl.google.com (dl.google.com)|142.251.37.46|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 103464324 (99M) [application/x-debian-package]
Saving to: 'google-chrome-stable_current_amd64.deb'

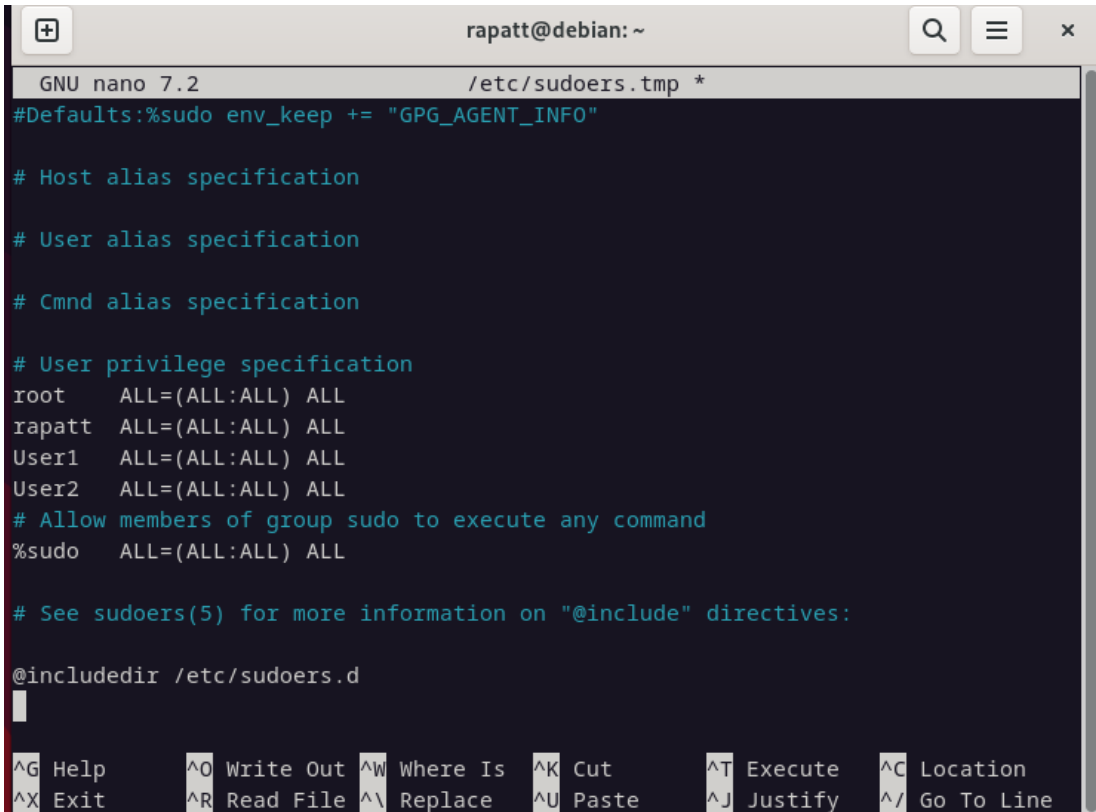
google-chrome-stable_curren 100%[=====>] 98.67M 36.5MB/s in 2.7s
2023-09-20 15:46:10 (36.5 MB/s) - 'google-chrome-stable_current_amd64.deb' saved [103464324/103464324]
root@debian:~#
```

Pour redémarrer notre machine nous faisons dans le terminal **reboot** et pour l'éteindre c'est **shutdown**.

Job 4

```
rapatt@debian:~$ su -  
Password:  
root@debian:~# EDITOR=nano visudo
```

Grâce à la commande `EDITOR=nano visudo` nous pouvons accéder au fichier sudoers.



```
rapatt@debian: ~  
GNU nano 7.2 /etc/sudoers.tmp *  
#Defaults:%sudo env_keep += "GPG_AGENT_INFO"  
  
# Host alias specification  
  
# User alias specification  
  
# Cmnd alias specification  
  
# User privilege specification  
root    ALL=(ALL:ALL) ALL  
rapatt  ALL=(ALL:ALL) ALL  
User1   ALL=(ALL:ALL) ALL  
User2   ALL=(ALL:ALL) ALL  
# Allow members of group sudo to execute any command  
%sudo   ALL=(ALL:ALL) ALL  
  
# See sudoers(5) for more information on "@include" directives:  
  
@includedir /etc/sudoers.d  
|  
  
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

Dans l'image ci-dessus nous retrouvons le fichier sudoers, nous avons rajouté dans ce fichier les User1 et User2 pour qu'ils puissent utiliser les commandes sudo.

```
rapatt@debian:~$ groupadd Plateformeurs  
bash: groupadd: command not found  
rapatt@debian:~$ sudo groupadd Plateformeurs  
[sudo] password for rapatt:  
rapatt@debian:~$ sudo useradd User1  
rapatt@debian:~$ sudo useradd User2  
rapatt@debian:~$ sudo usermod -a -G Plateformeurs User2  
rapatt@debian:~$ touch users.txt  
rapatt@debian:~$ touch droits.txt  
rapatt@debian:~$ nano  
rapatt@debian:~$ nano users.txt  
rapatt@debian:~$ cp users.txt droits.txt  
rapatt@debian:~$ cp users.txt groupes.txt
```

Nous pouvons voir sur l'image ci-dessus l'ajout du groupe Plateformeurs avec la commande **groupadd**, avec l'ajout des utilisateurs User1 et User2 avec la commande **useradd**.

Ensuite nous pouvons voir l'ajout de User2 dans le groupe Plateformeurs. Nous voyons aussi la créations des fichiers users.txt ainsi que droits.txt avec la commande **touch**.

Nous voyons aussi la copie de users.txt dans droits.txt et de users.txt dans groupes.txt grâce à la commande **cp**.

```
rapatt@debian:~$ sudo chown User1 droits.txt
```

Ici nous avons changé le propriétaire du fichier 1 pour que User1 soit le propriétaire avec la commande **chown**.

```
rapatt@debian:~$ sudo chmod 704 droits.txt
rapatt@debian:~$ sudo chmod 444 groupes.txt
rapatt@debian:~$ sudo chgrp Plateformeurs groupes.txt
```

Nous avons changé les droits du fichier droits.txt pour qu'il soit accessible en lecture uniquement et du fichier groupes.txt pour qu'il soit accessible en lecture uniquement par les utilisateurs et accessible en lecture écriture par le groupe Plateformeurs, tout ça grâce aux commandes **chmod** et **chgrp**.

Job 5

```

rapatt@debian:~$ alias la='ls -la'
rapatt@debian:~$ la
total 96
drwx----- 15 rapatt rapatt      4096 Sep 26 10:51 .
drwxr-xr-x  3 root   root      4096 Sep 20 13:05 ..
-rw-----  1 rapatt rapatt       622 Sep 26 10:45 .bash_history
-rw-r--r--  1 rapatt rapatt       220 Sep 20 13:05 .bash_logout
-rw-r--r--  1 rapatt rapatt     3526 Sep 20 13:05 .bashrc
drwx----- 11 rapatt rapatt      4096 Sep 20 14:54 .cache
drwx----- 11 rapatt rapatt      4096 Sep 20 15:10 .config
drwxr-xr-x  2 rapatt rapatt      4096 Sep 20 13:07 Desktop
drwxr-xr-x  2 rapatt rapatt      4096 Sep 20 13:58 Documents
drwxr-xr-x  2 rapatt rapatt      4096 Sep 20 13:07 Downloads
-rwx---r--  1 User1  rapatt       12 Sep 26 10:50 droits.txt
-rw-r--r--  1 rapatt rapatt     5290 Sep 20 13:05 .face
lrwxrwxrwx  1 rapatt rapatt        5 Sep 20 13:05 .face.icon -> .face
drwx-----  2 rapatt rapatt      4096 Sep 26 10:45 .gnupg
-r--r--r--  1 rapatt Plateformeurs 12 Sep 26 10:51 groupes.txt
drwx-----  4 rapatt rapatt      4096 Sep 20 13:07 .local
drwxr-xr-x  2 rapatt rapatt      4096 Sep 20 13:07 Music
drwxr-xr-x  2 rapatt rapatt      4096 Sep 20 13:07 Pictures
-rw-r--r--  1 rapatt rapatt       807 Sep 20 13:05 .profile
drwxr-xr-x  2 rapatt rapatt      4096 Sep 20 13:07 Public
drwx-----  2 rapatt rapatt      4096 Sep 20 13:10 .ssh
drwxr-xr-x  2 rapatt rapatt      4096 Sep 20 13:07 Templates
-rw-r--r--  1 rapatt rapatt       12 Sep 26 10:50 users.txt
drwxr-xr-x  2 rapatt rapatt      4096 Sep 20 13:07 Videos
rapatt@debian:~$ █

```

Dans les commandes ci-dessus nous voyons la création d'alias pour que la commande **la** fasse la même action que **ls -la**

```

rapatt@debian:~$ alias update='sudo apt-get update'
rapatt@debian:~$ update
Hit:1 http://deb.debian.org/debian bookworm InRelease
Get:2 http://security.debian.org/debian-security bookworm-security InRelease [48.0 kB]
Get:3 http://deb.debian.org/debian bookworm-updates InRelease [52.1 kB]
Fetched 100 kB in 1s (104 kB/s)
Reading package lists... Done
rapatt@debian:~$ █

```

Ici nous voyons l'alias **update** qui fait la même action que **sudo apt-get update**


```

rapatt@debian:~$ alias upgrade='sudo apt-get upgrade'
rapatt@debian:~$ upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  bind9-dnsutils bind9-host bind9-libs
3 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/2,113 kB of archives.
After this operation, 5,236 kB disk space will be freed.
Do you want to continue? [Y/n] y
Reading changelogs... Done
(Reading database ... 180570 files and directories currently installed.)
Preparing to unpack .../bind9-host_1%3a9.18.19-1~deb12u1_amd64.deb ...
Unpacking bind9-host (1:9.18.19-1~deb12u1) over (1:9.18.16-1~deb12u1) ...
Preparing to unpack .../bind9-dnsutils_1%3a9.18.19-1~deb12u1_amd64.deb ...
Unpacking bind9-dnsutils (1:9.18.19-1~deb12u1) over (1:9.18.16-1~deb12u1) ...
Preparing to unpack .../bind9-libs_1%3a9.18.19-1~deb12u1_amd64.deb ...
Unpacking bind9-libs:amd64 (1:9.18.19-1~deb12u1) over (1:9.18.16-1~deb12u1) ...
Setting up bind9-libs:amd64 (1:9.18.19-1~deb12u1) ...
Setting up bind9-host (1:9.18.19-1~deb12u1) ...
Setting up bind9-dnsutils (1:9.18.19-1~deb12u1) ...
Processing triggers for man-db (2.11.2-2) ...
Processing triggers for libc-bin (2.36-9+deb12u1) ...
rapatt@debian:~$

```

Dans l'image ci-dessus nous voyons l'alias de la commande **upgrade** qui fait la même action que la commande **sudo apt-get upgrade**

```

# some more ls aliases
#alias ll='ls -l'
alias la='ls -A'
alias update='sudo apt-get update'
alias upgrade='sudo apt-get upgrade'
USER='rapatt'
#alias l='ls -CF'

```

```

XDG_SESSION_CLASS=user
TERM=xterm-256color
USER=rapatt
GNOME_TERMINAL_SERVICE=:1.138

```

```
elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
fi
fi

export PATH="/home/$USER/Bureau"
rapatt@debian:~$
```

Dans les images ci-dessus nous voyons l'ajout de la variable d'environnement qui se nomme USER, nous voyons aussi la mise à jour des modifications de notre fichier .bashrc de notre shell actuel, nous voyons que notre Path est ajouter à notre chemin d'accès.

Job 6

```
rapatt@debian:~/Downloads$ tar xf Ghost\ in\ the\ Shell.tar
tar: Ignoring unknown extended header keyword 'LIBARCHIVE.xattr.com.apple.quarantine'
tar: Ignoring unknown extended header keyword 'LIBARCHIVE.xattr.com.apple.provenance'
tar: Ignoring unknown extended header keyword 'LIBARCHIVE.xattr.com.apple.macl'
rapatt@debian:~/Downloads$ ls
f2.tar  fichier.tar  'Ghost in the Shell.pdf'
rapatt@debian:~/Downloads$
```

Pour désarchiver la suite du sujet nous avons fait la commande **tar xf (nom du fichier).tar**, nous retrouvons bien la suite du sujet désarchiver en .pdf.

Job 7

```
rapatt@debian:~$ sudo echo "je suis votre fichier texte" > une_commande.txt && wc -l < /etc/apt/sources.list > nb_lignes.txt && ls -l /etc/apt/sources.list > save_sources && grep -r -o '\..*alias' / > fichiers_alias.txt
```

Nous pouvons voir que cette commande ci-dessus va exécuter en une ligne de commande toutes les actions demandées donc créer un fichier, écrire dedans... etc

```

rapatt@debian:~$ ls
Desktop    Downloads      Music          Pictures  save_sources  une_commande.txt
Documents  fichiers_alias.txt nb_lignes.txt Public     Templates     Videos
rapatt@debian:~$ cat une_commande.txt
je suis votre fichier texte
rapatt@debian:~$ cat fichiers_alias.txt
/home/rapatt/.bashrc:.bash_alias
/home/rapatt/.bashrc:.bash_alias
/home/rapatt/.bashrc:. ~/.bash_alias
/home/rapatt/.cache/gnome-software/odrs/ratings.json:.snapcraft.alias
/proc/dynamic_debug/control:.c:5673 [quirks]quirk_switchtec_ntb_dma_alias
/proc/dynamic_debug/control:.c:5696 [quirks]quirk_switchtec_ntb_dma_alias
/proc/dynamic_debug/control:.c:1972 [core]regulator_supply_alias
rapatt@debian:~$

```

Dans la photo ci-dessus nous pouvons voir que notre ligne de commande a bien créé tout ce qui était demandé dans le JOB 7 avec la création des fichiers demandés ainsi que la recherche des fichier commençant par. tout en cherchant le mot alias qui sera utilisé depuis un des fichiers.

Pour aller plus loin

Pour la fonction **tree** nous l'avons installé avec **sudo apt install tree**, nous lançons **tree** et nous sauvons les données dans le fichier **tree.save** en arrière plan grâce aux parenthèses. De plus nous avons **ls | wc -l** qui permet de lister et de compter le nombre d'éléments présents dans le dossier de courant. Et nous avons **sudo apt update** et **sudo apt upgrade** d'ailleurs l'upgrade ne marche pas si l'update n'a rien fait grâce aux **&&**.

```

sudo apt install tree && (tree / > tree.save &) || (ls | wc -l) && sudo apt update &&
(sudo apt upgrade || echo "L'update a échoué, l'upgrade ne sera pas lancé.")

```