

Tout savoir sur SHELL

Début

(toutes les phrases en bleu sont les commandes à mettre dans le shell Linux/Ubuntu)

Avant de commencer, tu es sur Linux ? Tu n'as pas besoin de changer les commandes ci-dessous. Tu es sur Windows ? Tu utiliseras toujours wsl avant les commandes par exemple **wsl ls -a**

Maintenant allons-y, pour commencer les commandes de base pour le shell Linux sont :

cd (passer dans un autre répertoire)
ls (lister le contenu d'un répertoire)
rm (supprimer un fichier ou un répertoire)
mv (déplacer un fichier ou un répertoire)
mkdir (créer un répertoire)
man (affiche un manuel)

Tu connais la plupart des commandes les plus utilisées, voici des exemples de comment les utiliser :

man ls (affichera le manuel de la commande ls)
ls -a (affiche les fichiers cachés du home de votre utilisateur)
ls -al (affiche les fichiers cachés pour les informations sur les droits sous forme de liste)

Les - sont appelés options, c'est quoi une option ? Et bien, ce sont des paramètres modifiant le script ou le comportement du shell.

Il y a deux types d'options :

- un simple tiret est une option courte
- un double tiret est une option longue et signifie la fin des options.

Lire un fichier est super simple, il faut écrire **less .nomdufichier**

Tu veux lire que certaines lignes, ce n'est pas less que tu dois utiliser mais head pour lire le début d'un fichier et tail pour lire la fin d'un fichier.

head -10 .bashrc pour les 10 premières lignes
head -20 .bashrc pour les 20 premières lignes
tail -10 .bashrc pour les 10 dernières lignes
tail -20 .bashrc pour les 20 dernières lignes

Les paquets

Un paquet est un bout de logiciel prêt à être installé où on peut établir nos besoins, nos compatibilités et nos incompatibilités.

Pour installer un paquet on doit faire **sudo apt install mon_paquet** mais ici on veut installer cmatrix donc **sudo apt install cmatrix**
Maintenant pour lancer le paquet que tu viens d'installer on fait **sudo mon_paquet** donc ici **sudo cmatrix**

Pour mettre à jour son gestionnaire de paquets on fait **sudo apt update**
et pour mettre à jour ses différents logiciels **sudo apt upgrade**

Essayons d'installer Google Chrome avec shell :

On installe le paquet wget : **sudo apt install wget**

On utilise wget pour télécharger chrome avec le lien : **sudo wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb**

On va unpacking Google chrome : **sudo dpkg -i google-chrome-stable_current_amd64.deb**

Et pour finir on utilise cette commande pour rectifier toutes les erreurs lors du téléchargement : **sudo apt-get install -f**

Si tu veux directement installer la page google puis l'ouvrir depuis ton navigateur par défaut on utilisera :

sudo wget www.google.com

x-www-browser index.html

Et voilà, cela ouvrira ta page google

Eteignons et redémarrons le pc à l'aide de shell

Eteindre : **sudo halt**

Redémarrer : **sudo reboot**

Droits, propriétaires et groupes

On va tout faire grâce au Shell Ubuntu 👍

Création du fichier users.txt en mettant User1 et User2 :

cat > users.txt

User1

User2

Pour créer un groupe : **sudo groupadd Plateformeurs**

Maintenant il faut créer les utilisateurs et mettre dans Plateformeurs :

sudo useradd User1

sudo useradd User2

pour mettre dans un groupe on fait **sudo adduser User2 Plateformeurs**

Copions le fichier users.txt qu'on a préalablement créé et crée une copie qui s'appellera droits.txt et groupes.txt soit **cp users.txt droits.txt** et **cp users.txt groupes.txt**

Pour changer le propriétaire d'un fichier ici on va prendre droits.txt comme exemple, on aura ceci **sudo chown User1 droits.txt**

Pour les droits, on a 3 types d'utilisation : u (propriétaire/user), g (groupes) et o (pour les autres) puis les droits on a - pour enlever + pour ajouter et = pour définir les droits.

Donc maintenant on doit définir les droits, changeons les droits pour que User2 ai accès seulement en lecture dans le fichier droits.txt.

Comme le propriétaire du fichier est User 1, on utilisera o soit **sudo chmod o=r droits.txt**

On doit maintenant définir que les utilisateurs ont accès au lecture uniquement au fichier groupes.txt soit **sudo chmod u=r groupes.txt**

Et pour le dernier fichier, le groupe "Plateformeurs" doit avoir accès à la lecture et écriture du fichier 'groupes.txt' soit **sudo chmod g=rw groupes.txt**

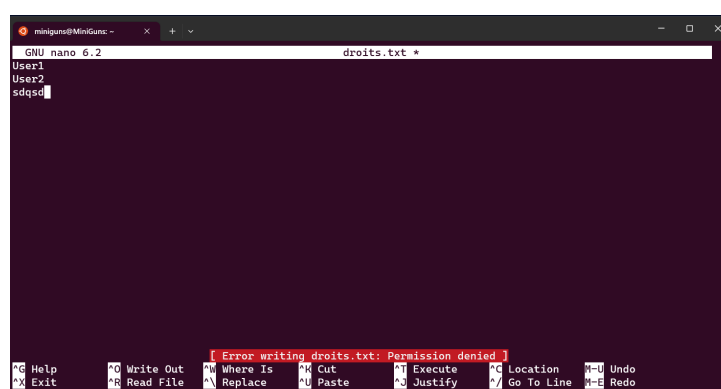
Pour vérifier tout ça, on utilise **ls -l** pour avoir accès au fichier avec les droits, propriétaires ...

Et pour tester on entre d'abord dans un utilisateur donc essayons avec User2 : **su User2**

enfin on entre dans le fichier droits.txt pour tester, ici User2 n'a pas le droit de modifier le texte normalement

on utilise **nano droits.txt**

et maintenant tentons d'ajouter quelque chose dans le fichier et bien on voit que rien ne marche on n'a pas la permission. Parfait tout est bon



```
miniguns@Miniguns: ~  
GNU nano 6.2 droits.txt *  
User1  
User2  
sdqsd  
[Error writing droits.txt: Permission denied]  
^G Help  ^O Write Out  ^W Where Is  ^K Cut        ^T Execute   ^C Location  ^H-U Undo  
^X Exit  ^R Read File  ^A Replace   ^U Paste      ^J Justify   ^V Go To Line ^H-B Redo
```

Changement du fichier .bashrc

On va maintenant aborder les alias c'est assez simple comme du python, on ajoute une variable donc on commence par **alias la='ls -la'** qui lancera la commande de **ls -la** en tapant que **la**. On va faire de même pour apt-get update et apt-get upgrade donc on **alias update='apt-get update'** et **alias upgrade='apt-get upgrade'**

On va ajouter une variable d'environnement pour ce faire on fait **export USER=nom_d'utilisateur**

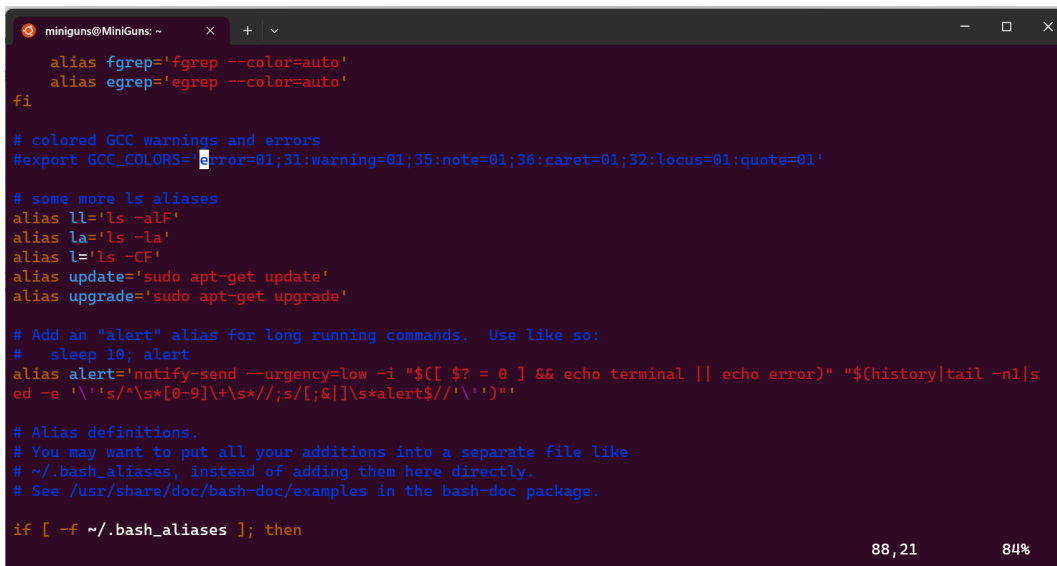
Pour mettre à jour les modifications de ton bashrc dans notre shell actuel il faut plusieurs étapes, tout d'abord on ouvre .bashrc en faisant **vi .bashrc**

On ajoute les alias et la variable d'environnement indiqué

Pour quitter on appuie sur échap et puis on écrit **:x**

Ensuite on va sourcer donc **source ~/.bashrc**

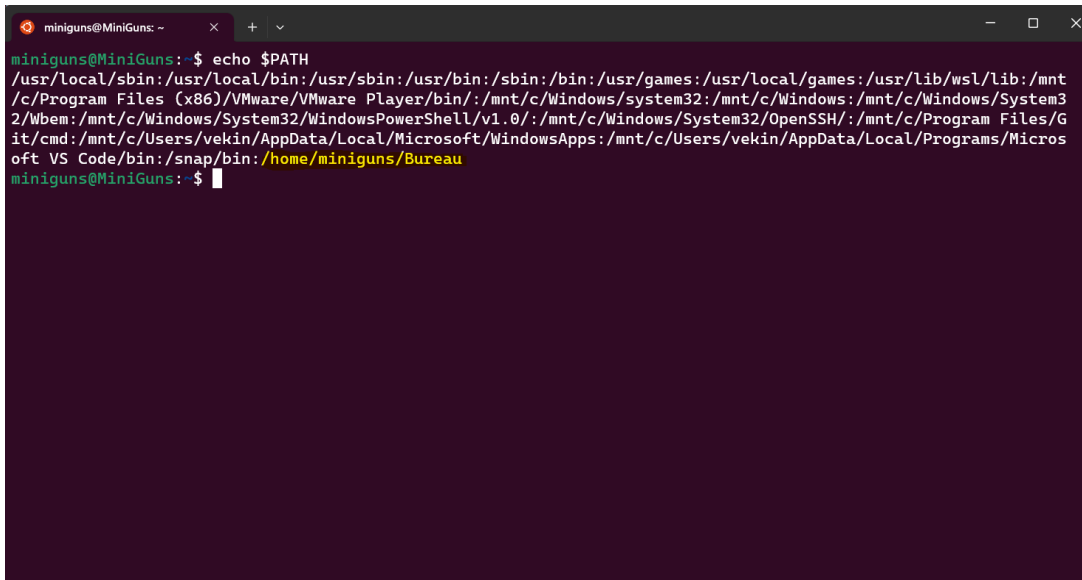
et enfin **exec bash** voilà maintenant on a juste à ouvrir un autre terminal et tout fonctionnera à merveille



```
miniguns@MiniGuns: ~  
alias fgrep='fgrep --color=auto'  
alias egrep='egrep --color=auto'  
fi  
  
# colored GCC warnings and errors  
#export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'  
  
# some more ls aliases  
alias ll='ls -aF'  
alias la='ls -la'  
alias l='ls -CF'  
alias update='sudo apt-get update'  
alias upgrade='sudo apt-get upgrade'  
  
# Add an "alert" alias for long running commands. Use like so:  
#   sleep 10; alert  
alias alert='notify-send --urgency=low -i "${[ $? = 0 ]} && echo terminal || echo error)" "${history|tail -n1|sed -e '\''s/^s*[0-9]\+\s*//;s/[;&]\s*alert$//'\''}"'  
  
# 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
```

Pour afficher les variables d'environnements c'est assez simple on utilise juste **printenv**

Ajoutons maintenant dans PATH le chemin
/home/nom_utilisateur/Bureau pour ce faire il faut taper la commande
export PATH=\$PATH:/home/votre_utilisateur/Bureau
et pour vérifier que tout est bon **echo \$PATH** on aura du coup notre path
comme ci dessous



```
miniguns@MiniGuns: ~  
miniguns@MiniGuns:~$ echo $PATH  
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/usr/lib/wsl/Lib:/mnt  
/c/Program Files (x86)/VMware/VMware Player/bin:/mnt/c/Windows/system32:/mnt/c/Windows:/mnt/c/Windows/System3  
2/Wbem:/mnt/c/Windows/System32/WindowsPowerShell/v1.0/:/mnt/c/Windows/System32/OpenSSH/:/mnt/c/Program Files/G  
it/cmd:/mnt/c/Users/vekin/AppData/Local/Microsoft/WindowsApps:/mnt/c/Users/vekin/AppData/Local/Programs/Micros  
oft VS Code/bin:/snap/bin:/home/miniguns/Bureau  
miniguns@MiniGuns:~$
```

Désarchiver un .tar

Désarchiver un dossier ? assez simple on écrit juste

tar -xvzf GhostintheShell.tar

Il faut bien sur être répertorié dans l'endroit exact où est le fichier .tar
en faisant **cd endroit** et **ls**

Tout en un

Le but de l'exercice est de créer un fichier et mettre un texte puis compter
le nombre de ligne du fichier source apt et le mettre dans un fichier
nb_lignes.txt ensuite afficher le contenu du fichier source apt et
l'enregistrer dans un autre fichier save_sources et enfin faire une
recherche des fichiers commençants par '.' tout en cherchant le mot alias
qui sera utilisé dans un fichier

voici la commande en une seule ligne :

```
echo 'Je suis votre fichier texte' > une_commande.txt && wc -l  
/etc/apt/sources.list > nb_lignes.txt && less /etc/apt/sources.list >  
save_sources && ls -a && grep "alias" .*
```

Tree (pour aller plus loin)

Dernier point qu'on va aborder sur la documentation, on va réunir toutes les informations qu'on a appris et utilisé ci dessous le but installer tree, lancer tree en arrière plan et enregistrer dans un fichier tree.save puis lister les éléments de ls pour y compter le nombre de lignes et pour finir on va update et upgrade en une seule commande. Pour ceci on fera :

```
sudo apt install tree && tree / > tree.save & ls & ls | wc -l && sudo apt  
update && sudo apt upgrade
```