

Shell

Le shell est une couche logiciel qui fournit l'interface utilisateur d'un système d'exploitation. C'est un programme qui reçoit des commandes informatiques données par un utilisateur à partir du clavier pour les envoyer au système d'exploitation qui se chargera de les exécuter. Le Shell peut prendre deux formes :

- Sans interface graphique (CLI), permet à l'utilisateur d'interagir avec le système à partir de commandes adaptées en mode texte, ce qui permet par exemple l'installation de logiciel ou d'application sans environnement graphique.
- Avec interface graphique (GUI), tous les ordinateurs de nos jours en ont un, un environnement de bureau ou d'écran d'accueil.

On peut utiliser le Terminal pour dialoguer avec le système d'exploitation.

C'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 les résultats des commandes.

Job 1

La commande `man` permet d'afficher le manuel d'une autre commande. Par exemple `man ls` permet d'afficher les options de la commande `ls`, c'est une commande disponible sur les systèmes d'exploitation.

Les fichiers cachés sont reconnus par le point précédent leur nom, exemple `“.fichier”` sera reconnu comme un fichier caché alors que `“fichier”` ne sera pas reconnu, donc pour afficher les fichiers cachés du home de l'utilisateur on tape la commande `ls -d`. Pour afficher les droits sous forme de liste, la commande `ls -dl` fera l'affaire.

Pour ajouter une ou plusieurs options à une commande sur un terminal il faut ajouter un **tiret “-”**. Le tiret fait partie d'une des deux principales syntaxes d'écriture des options pour une commande, la deuxième la plus répandue est le **double tiret “--”**. Le premier mot tapé avant le ou les tirets est une commande et celle placée après sont des options, tous le reste sont des paramètres.

- Afficher le manuel de la commande `ls`: `“man ls”`
- Afficher les fichiers cachés du home de votre utilisateur : `“ls -d”`
- Afficher les fichiers cachés plus les informations sur les droits sous forme de liste: `“ls -dl”`
- Ajouter des options à une commande: **un tiret “-”**
- Les deux syntaxes principales d'écriture des options pour une commande: **le tiret “-” et le double tiret “--”**

Job 2

Il existe plusieurs commandes couramment utilisées le contenu d'un fichier, celle qu'on utilise ici est **"cat"**, il faudra donc taper cat puis le nom du fichier précédé par un point **"cat .bashrc"**. Par contre, lorsque ce sont des parties spécifiques du fichier qui doivent être lues, ce sont les commandes **"head"** et **"tail"** que nous utiliserons. La commande **"head"** permettra par exemple d'afficher les 10 premières lignes du fichier en tapant la commande puis le nom du fichier précédé par un point **"head .bashrc"**. Alors que la commande **"tail"** permet d'afficher les 10 dernières lignes d'un fichier en tapant la commande puis le nom du fichier précédé par un point **"tail .bashrc"**. Si votre fichier est assez volumineux et que vous ne voulez pas rester seulement aux 10 premières lignes et en lire par exemple 20 il faudra rajouter l'option **"-n20"** entre la commande et le nom du fichier et pareillement pour lire les 20 dernières lignes.

- Lisez un fichier en utilisant une commande qui permet seulement de lire: **"cat .bashrc"**
- afficher les 10 premières lignes du fichier **".bashrc"**: **"head .bashrc"**
- afficher les 10 dernières lignes du fichier **".bashrc"**: **"tail .bashrc"**
- afficher les 20 premières lignes du fichier **".bashrc"**: **"head -n20 .bashrc"**
- afficher les 20 dernières lignes du fichier **".bashrc"**: **"tail -n20 .bashrc"**

Job 3

Cmatrix est un utilitaire qui permet d'afficher la Matrix directement sur le terminal, il ne sert pas à grand chose mais c'est une bonne première étape pour apprendre à installer des paquets. On tape alors **sudo apt install cmatrix**, ici **sudo** permet à l'utilisateur d'exécuter plusieurs commandes en tant que super utilisateur pour pouvoir installer le paquet dans le terminal sans être bloqué. Vu que le paquet provient d'un dépôt APT on utilise donc l'outil en ligne de commande **apt** qui permet principalement d'installer et de désinstaller des logiciels en ligne de commande. Pour lancer le paquet qu'on vient de taper il suffit juste de taper le fichier **cmatrix** sur le terminal.

- Installer le paquet **"cmatrix"**: **"sudo apt install cmatrix"**
- lancer le paquet que vous venez d'installer: **"cmatrix"**
- Mettre à jour son gestionnaire de paquets: **"apt-get update"**
- Mettre à jour ses différents logiciels: **"apt-get upgrade"**

Google Chrome n'est pas une application présente dans les dépôts officiels. Elle ne bénéficie ni de la validation officielle Ubuntu, ni du support des développeurs des équipes officielles Ubuntu. Installer une application hors des dépôts officiels peut présenter des risques d'instabilité de votre système. Heureusement pour nous cette méthode nous permet d'installer la version stable de google chrome directement sur le terminal sans soucis. On

utilise la commande `"sudo apt install google-chrome-stable"` c'est la version de l'application qui est la plus sûre et qui est souvent mise à jour.

- Télécharger les internets : Google `"sudo apt install google-chrome-stable"`

Pour que tous les lancements de l'application se font correctement on nous propose de redémarrer le système, pour cela on utilise la commande `"reboot"`. Comme tout système d'exploitation, il peut y avoir des problèmes avec des pilotes ou des applications. Utiliser la commande `"power off"` pour éteindre la machine.

- Redémarrer votre machine: `"reboot"`
- éteindre votre machine: `"power off"`

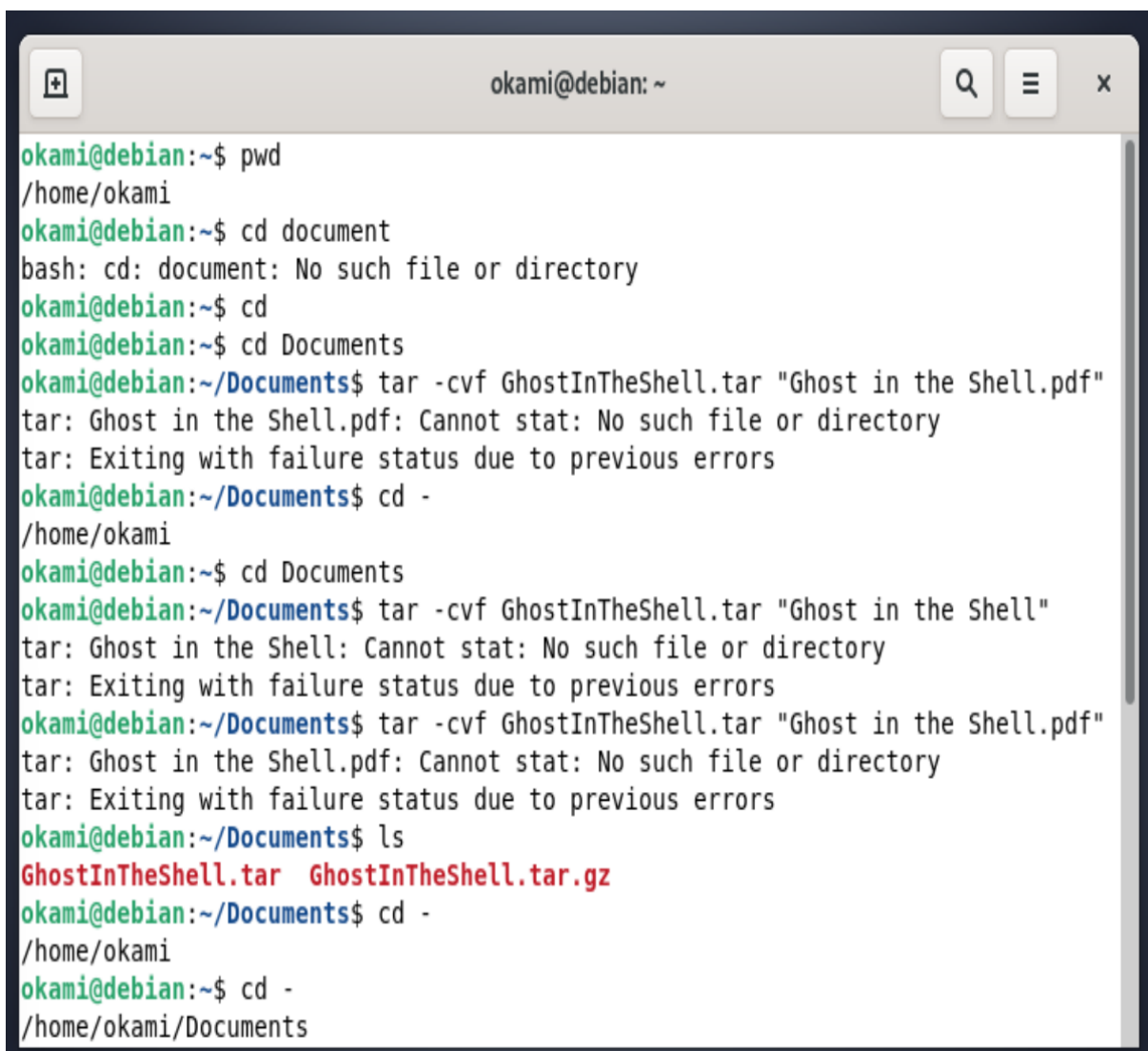
Job 4

- Créer un fichier `users.txt` qui contiendra User1 et User2 séparé par un retour à la ligne: `"rm users.txt"` puis `"cat > users.txt"`
- Créer un groupe appelé "Plateformeurs": `"sudo groupadd plateformeurs"`
- Créer un utilisateur appelé "User1": `"sudo adduser user1"`
- Créer un utilisateur appelé "User2": `"sudo adduser user2"`
- Ajouter "User2" au groupe Plateformeurs: `"sudo groupadd user2 plateformeurs"`
- Copier votre `"users.txt"` dans un fichier `"droits.txt"`: `"cp users.txt droits.txt"`
- Copier votre `"users.txt"` dans un fichier `"groupes.txt"`: `"cp users.txt droits.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 740 droits.txt"`
- Changer les droits du fichier `"groupes.txt"` pour que les utilisateurs puissent accéder au fichier en lecture uniquement: `"sudo chmod 744 groupes.txt"`
- Changer les droits du fichier pour que le groupe "Plateformeurs" puissent y accéder en lecture/écriture. `"sudo chmod 760 groupes.txt"`

Job 5

- Ajouter un alias qui permettra de lancer la commande "ls -la" en tapant "la": `alias la='ls -la'`
- Ajouter un alias qui permettra de lancer la commande "apt-get update" en tapant "update": `alias update='apt-get update'`
- Ajouter un alias qui permettra de lancer la commande "apt-get upgrade" en tapant "upgrade": `alias upgrade='apt-get upgrade'`
- Ajouter une variable d'environnement qui se nommera "USER" et qui sera égale à votre nom d'utilisateur: `export USER=(utilisateur)`
- Mettre à jour les modifications de votre bashrc dans votre shell actuel: source .bashrc
- Afficher les variables d'environnement: `printenv`
- Ajouter à votre Path le chemin "/home/votre utilisateur/Bureau": `export path=spath`

Job 6



```
okami@debian: ~  
okami@debian:~$ pwd  
/home/okami  
okami@debian:~$ cd document  
bash: cd: document: No such file or directory  
okami@debian:~$ cd  
okami@debian:~$ cd Documents  
okami@debian:~/Documents$ tar -cvf GhostInTheShell.tar "Ghost in the Shell.pdf"  
tar: Ghost in the Shell.pdf: Cannot stat: No such file or directory  
tar: Exiting with failure status due to previous errors  
okami@debian:~/Documents$ cd -  
/home/okami  
okami@debian:~$ cd Documents  
okami@debian:~/Documents$ tar -cvf GhostInTheShell.tar "Ghost in the Shell"  
tar: Ghost in the Shell: Cannot stat: No such file or directory  
tar: Exiting with failure status due to previous errors  
okami@debian:~/Documents$ tar -cvf GhostInTheShell.tar "Ghost in the Shell.pdf"  
tar: Ghost in the Shell.pdf: Cannot stat: No such file or directory  
tar: Exiting with failure status due to previous errors  
okami@debian:~/Documents$ ls  
GhostInTheShell.tar  GhostInTheShell.tar.gz  
okami@debian:~/Documents$ cd -  
/home/okami  
okami@debian:~$ cd -  
/home/okami/Documents
```

La commande **Tar** de Linux est l'une des plus utilisées pour la compression, archiver, désarchiver, ect...

Tar est l'abréviation de Tape archive et sert à compresser une collection de fichiers et de dossiers.

Dans la plupart des cas, une fois la compression effectuée à l'aide de tar, on obtient un fichier .tar. Une compression plus poussée est effectuée à l'aide de gzip, ce qui donne un fichier .tar.gz.

Avec tar, vous pouvez compresser et décompresser des fichiers.

Job 7

- Créer un fichier "une commande.txt" avec le texte suivant "Je suis votre fichier texte": **"echo "je suis votre fichier"> une_commande.txt"; pour vérifier : cat une_commande.txt**
- Compter le nombre de lignes présentes dans votre fichier de source apt: **"cat /etc/apt/sources.list | wc -l"** et les enregistrer dans le fichier nommé "nb_lignes.txt": **"cat -n /etc/apt/sources.list > nb_lignes.txt"**
- Afficher le contenu du fichier source apt et l'enregistrer dans un autre fichier appelé **"save_sources"**: **"cat sources.list> save_sources"**

Pour aller plus loin..

- Installer la commande tree: **"sudo apt-get install tree"**
- 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" puis "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 | wc -l"**

- Lancer une commande pour update vos paquets, si l'update réussi alors, vous devrez lancer un upgrade de vos paquets. Si l'update échoue, votre upgrade ne se lancera pas: d'abord passer en route "su -" tapez votre mot de passe puis la commande **"apt update && apt upgrade"**