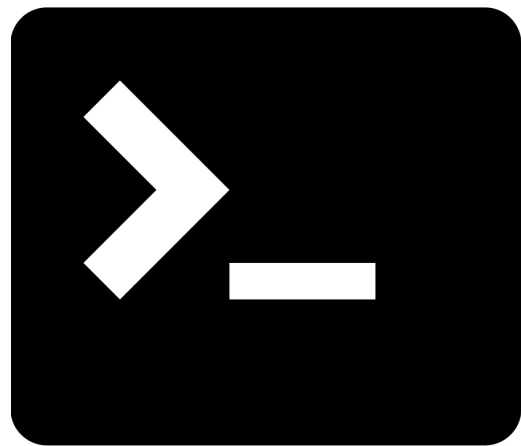


Shell

By Catherine Tranchand



Définitions

Terminal

Un terminal est un environnement d'entrée et de sortie.

Par exemple en entrée un clavier et en sortie un écran ou une imprimante.

Shell

Le shell a toujours été présent – à partir de l'aube d'UNIX.

Le Shell est **l'interpréteur en ligne de commande**, il permet à un utilisateur de s'identifier sur une console ou un terminal. Il fournit un certain nombre de commandes pour interagir avec le système d'exploitation.

En général, il permet aussi de créer des fichiers script que l'on nomme script pour exécuter plusieurs commandes l'une après l'autre. C'est donc aussi un langage de programmation.

- Sur Linux : Bourne shell (sh), KornShell (ksh), C shell (csh), tcsh, bash, Z shell (zsh)
- Sur Windows : PowerShell

La différence entre terminal, console et shell:

- Terminal = TTY : Environnement d'entrée / sortie texte
- Console : terminal physique
- Shell : Interprète ligne de commande

Job 1

Votre fichier de documentation contiendra les actions ci-dessous ainsi que leur équivalent en ligne de commande Linux :

- Afficher le manuel de la commande **ls**
- Afficher les fichiers cachés du home de votre utilisateur
- Afficher les fichiers cachés plus les informations sur les droits sous forme de

liste

man ls Pour afficher la manuel de la commande **ls (list)**

ls -a Pour afficher tous les fichiers et les fichiers cachés. Le paramètre *-a* permet d'inclure des fichiers cachés

ls -a -l Pour afficher les infos sur les droits

Questions:

Comment ajouter des options à une commande ?

Le 1er mot tapé est une commande. Les lettres tapées après un tiret et les mots tapés après deux tirets, sont des options. Donc pour ajouter une option à une commande il faut tout simplement faire un tiret plus l'option (représentée par une lettre).

Exemples:

ls -a

ls -all

shutdown -r

shutdown -h

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

Pour une commande avec un seul paramètre, il suffit d'un d'entrer la commande + un flag et le paramètre.

Lorsqu'on utilise un paramètre avec plus d'une lettre il faudra écrire la commande + - - le paramètre.

Job 2

Votre fichier de documentation contiendra les actions ci-dessous ainsi que leur équivalent en ligne de commande Linux :

- Lisez un fichier en utilisant une commande qui permet seulement de lire
- afficher les 10 premières lignes du fichier “.bashrc”
- afficher les 10 dernières lignes du fichier “.bashrc”
- afficher les 20 premières lignes du fichier “.bashrc”
- afficher les 20 dernières lignes du fichier “.bashrc”

Pour MacOS - Z Shell

Cat est une commande qui permet de lire un fichier.

cat .zshrc / less .zshrc Lecture du fichier “.zshrc”. Commande cat +nom_du_fichier

Head est une commande qui permet d'afficher les premières lignes de texte d'un fichier ou de l'entrée standard. La syntaxe de cette commande est la suivante :

head [options] <fichier>

head .zshrc Pour afficher 10 premières lignes du fichier “.zshrc”.

Par défaut, **head** affiche les 10 premières lignes de son entrée sur la sortie standard. Le nombre de lignes affichées peut être changé avec une option fournie sur la ligne de commande. L'exemple suivant affiche les 20 premières lignes de *fichier* :

head -n20 .zshrc Pour afficher 20 premières lignes. Option **-n** indique le nombre des lignes à afficher.

La commande tail permet à un utilisateur d'afficher la fin d'un fichier

tail -n20 .zshrc Pour afficher 20 dernières lignes .

Job 3

Votre fichier de documentation contiendra les actions ci-dessous ainsi que leur équivalent en ligne de commande Linux :

- Installer le paquet “**cmatrix**”
- lancer le paquet que vous venez d’installer
- Mettre à jour son gestionnaire de paquets
- Mettre à jour ses différents logiciels
- Télécharger les internets : Google
- Redémarrer votre machine
- éteindre votre machine

Pour installer le paquet **cmatrix**, il faut passer d’abord le gestionnaire du paquet qui est déjà installé sur Linux. La commande: **sudo apt install cmatrix** ou bien **sudo apt-get install cmatrix**.

Pour faire cet exercice, j'utilise MacOS. Pour mon OS il faut d’abord installer le gestionnaire de paquets *homebrew*.

Voici mon chemin:

Installer “Cmatrix”

1. D’abord il faudra installer **Homebrew**

Homebrew gère des paquets pour MacOS. Pour l’installer il faut exécuter la commande ci-dessus (copier-coller) et taper *enter*

```
$ /usr/bin/ruby -e "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

- Maintenant il faut fermer et rouvrir le terminal. La commande *brew* est désormais disponible dans le terminal.
- Pour installer CMatrix: `$ brew install cmatrix`
- Installer et Configurer GNU Screen

```
$ brew install homebrew/homebrew-core/screen
```

Ensuite il faudra créer un fichier *screenrc* afin d'y placer la configuration de notre plugin. Pour créer un fichier commande *touch*. Le tilde servira à placer le fichier au niveau de notre répertoire Home.

```
$ touch ~/.screenrc
```

Les fichiers commençant par “.” sont les fichiers cachés. Pour vérifier que le fichier s'est bien créé, on peut taper la commande *ls -la*

- En tapant dans le shell *screen* *cmatrix* va s'afficher:

brew update Pour mettre le gestionnaire de paquet *homebrew* sur mon Mac à jour

sudo apt update Pour mettre à jour le gestionnaire de paquet sur Linux.

Pour mettre à jour les différents logiciels : **sudo apt update** sur Linux /

softwareupdate -i -a sur macOS. L'option -i correspond à *install*, -a à *all*. Pour Linux **sudo apt upgrade**.

Pour télécharger Google il faut passer par le gestionnaire de paquet **wget** (Wget est un programme en ligne de commande non interactif de téléchargement de fichiers depuis le Web). Wget n'est pas préinstallé sur MacOS donc je l'installe d'abord: **brew install wget**.

Maintenant je peux télécharger Google: **wget -r install www.google.com** (option -r active le téléchargement récursif.)

Pour redémarrer l'ordinateur: **sudo shutdown -r now** (option -r active le redémarrage). Cette commande ne peut être exécutée que en mode super user.

Pour éteindre l'ordinateur: **sudo shutdown -h now** (option -h permet d'éteindre l'ordinateur)

Pour Linux vous avez plusieurs possibilités pour éteindre la machine: halt, poweroff, shutdown

Job 4

Votre fichier de documentation contiendra les actions ci-dessous ainsi que leur équivalent en ligne de commande Linux :

- Créer un fichier **users.txt** qui contiendra **User1** et **User2** séparé par un retour à la ligne
- Créer un groupe appelé **“Plateformeurs”**
- Créer un utilisateur appelé **“User1”**
- Créer un utilisateur appelé **“User2”**
- Ajouter **“User2”** au groupe **Plateformeurs**
- Copier votre **“users.txt”** dans un fichier **“droits.txt”**
- Copier votre **“users.txt”** dans un fichier **“groupes.txt”**
- Changer le propriétaire du fichier **“droits.txt”** pour mettre **“User1”**
- Changer les droits du fichier **“droits.txt”** pour que **“User2”** ai accès seulement en lecture ok
- Changer les droits du fichier **“groupes.txt”** pour que les utilisateurs puissent accéder au fichier en lecture uniquement
- Changer les droits du fichier pour que le groupe **“Plateformeurs”** puissent y accéder en lecture/ écriture

Pour créer un fichier il suffit de taper touch + nom de fichier: **touch ~/.users.txt** (touch - pour créer des fichiers vides, la signe "~" indique que votre fichier va être placé dans *home*. Pour éditer le fichier: **nano .users.txt**

Ici je peux noter: **User1**

User2

Ctrl+x - pour quitter le mode nano

Pour créer un groupe sur Linux il suffit de faire **sudo addgroup + nom_du_groupe**

Pour créer un utilisateur sur Linux il suffit de faire **sudo adduser + nom_utilisateur**

Sur macOS, **sudo dscl** espace "." espace, puis -create et le chemin : **sudo dscl . -create /groups/nom_groupe.**

Il faut penser à bien ajouter un GID pour le groupe (ID de groupe) :

sudo dscl . -create /Groups/Plateformeurs gid 333

Pour lister les groupes sur Linux il faut taper la commande **cat /etc/group.**

Pour lister les groupes sur MacOS il faut taper la commande : **dscl . -list /groups**

Pour créer un nouvel utilisateur, on fera la même commande que précédemment : **sudo dscl . -create /Users/user2**

L'on peut ajouter un UniqueID à chaque utilisateur afin de faciliter l'ajout de l'utilisateur dans un groupe avec la syntaxe suivante : **sudo dscl . -create /Users/user1 UniqueID 6666**

Pour bien vérifier les utilisateurs : **dscl . -list /Users UniqueID**

Si je souhaite ajouter un utilisateur, ici user2, au groupe Plateformeurs j'entre la commande :

sudo dseditgroup -o edit -a user2 -t user Plateformeurs

La commande *dseditgroup* est un outil de manipulation et d'enregistrement des groupes. Le paramètre -o renvoie à l'opération que l'on peut spécifier (edit, read, delete, checkmember.) -a sert à spécifier le nom de l'utilisateur à ajouter, le -t renvoie au type qui va être ajouté.

Pour vérifier que l'utilisateur est dans le groupe : **dscacheutil -q group -a name Plateformeurs**

La commande *dscacheutil* sert à recueillir des informations, des statistiques.

L'option **-q** renvoie à la catégorie, le **-a** au nom du groupe.

Pour supprimer un utilisateur d'un groupe : **sudo dscl . -delete /Groups/Plateformeurs GroupMembership user2**

Pour copier un fichier dans un autre fichier : **cp users.txt droits.txt**

Pour changer le propriétaire d'un fichier : **sudo chown 6666(id de mon user) droits.txt**

Pour vérifier **ls -l**

Pour changer les droits d'un utilisateur sur le fichier il faut utiliser la commande **chmod**

-> pour que le propriétaire ait tous les droits, que les membres d'un groupe et les autres utilisateurs n'aient que le droit de lecture on tapera donc la commande : **sudo chmod 744 droits.txt** (ce n'est pas un exécutable donc on pourrait mettre 644)

Pour changer le groupe d'un fichier il faut taper la commande : **sudo chown user1:Plateformeurs groupes.txt**

Il faut ensuite autoriser le groupe à lire uniquement via la commande **sudo chmod 744 groupes.txt**

On peut ainsi vérifier que le groupe Plateformeurs a les droits d'écriture et de lecture avec **dscacheutil -q group -a name Plateformeurs**

Pour changer les droits d'un utilisateur sur le fichier, il faut utiliser la commande *chmod (change file modes or access control lists)*. Pour que le propriétaire ait tous les droits, que les membres d'un groupe et les autres utilisateurs n'aient que le droit de lecture on tapera donc la commande : **sudo chmod 744 droits.txt** (ce n'est pas un exécutable donc on pourrait mettre 644 si l'on voulait). Chaque chiffre représente un droit, l'accumulation de ces chiffres peuvent amener à 7 ce qui représente tous les droits. Il y a donc deux façon pour connaître ou modifier les droits, par les lettres r,w,x ou par les chiffres, cela se nomme la valeur octale.

Pour changer le propriétaire du fichier et affilier un groupe à ce fichier là j'entre dans le *shell* **sudo chown 6666:Plateformeurs groupes.txt**

Pour faire un calcul et gagner du temps ::

<https://www.cactus2000.de/fr/unit/massunx.shtml>

Job 5

Votre fichier de documentation contiendra les actions ci-dessous ainsi que leur équivalent en ligne de commande Linux :

- Ajouter un alias qui permettra de lancer la commande “ls -la” en tapant “la”
- Ajouter un alias qui permettra de lancer la commande “apt-get update” en tapant “update”
- Ajouter un alias qui permettra de lancer la commande “apt-get upgrade” en tapant “upgrade”
- Ajouter une variable d’environnement qui se nommera “USER” et qui sera égale à votre nom d’utilisateur
- Mettre à jour les modifications de votre bashrc dans votre shell actuel
- Afficher les variables d’environnement
- Ajouter à votre Path le chemin “/home/'votre utilisateur'/Bureau”

Un **alias** est un raccourci vers une app, un fichier ou un dossier que vous pouvez conserver à n’importe quel emplacement sur votre Mac. Lorsque vous créez un alias pour un élément, l’élément réel n’est pas déplacé. Lorsque vous ouvrez un alias, l’élément d’origine s’ouvre. Si vous déplacez ou renommez l’élément d’origine, cela n’a aucune incidence sur son lien avec l’alias.

Les alias

Pour faire un alias de ls -la il suffit d’écrire **alias l='ls -la'**

Pour l’alias du gestionnaire de paquet sur macOS **alias update='brew update'**

Pour l’alias du tout **alias upgrade='brew upgrade'**

Même méthode pour Linux mais les commandes de base sont *apt-get update* et *apt-get upgrade*.

Variables d’environnement

Pour créer une variable d’environnement temporaire sur macOS :

export USER=user1 USER

ou

export USER=EkaterinaTranchand

Pour voir toutes les variables : **env**

Pour ajouter un chemin dans une variable d'environnement qui n'est pas temporaire il faut aller dans le fichier .zshrc et rajouter le chemin dans la variable souhaitée.

Il faut penser à mettre à jour son shell avec source ~/.zshrc pour que les aliases et les variables soient prise en compte.

Pour ajouter à mon Path le chemin "/home/'votre utilisateur'/Bureau":

export PATH=home/EkaterinaTranchand/Desktop

Pour vérifier : **echo \$PATH**

Job 6

Je n'ai pas réussi à faire cet exercice...

Job 7

Toutes les actions sont à réaliser en une seule commande

Maintenant, vous allez approfondir les commandes, avec les caractères suivants "> < >> <<|", votre fichier de documentation contiendra les actions ci-dessous ainsi que leur équivalent en ligne de commande Linux :

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

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

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

-Faites une recherche des fichiers commençants par "." tout en cherchant le mot alias

qui sera utilisé depuis un fichier

Pour créer un fichier texte `une_commande.txt` tout en écrivant un texte au sein de ce fichier, on tape la commande `echo "texte que l'on souhaite écrire" > une_commande.txt`

echo "Je suis votre fichier texte">une_commande.txt

Pour vérifier que le texte est bien dans le fichier: **cat une_commande.txt**

La commande **wc** de Linux vous permet de compter le nombre de lignes, de mots, de caractères et d'octets de chaque fichier donné ou d'entrée standard et d'imprimer le résultat.

Options	Description
-l, --lines	Imprimez le nombre de lignes
-w, --words	Imprimez le nombre de mots
-m, --chars	Imprimez le nombre de caractères
-c, --bytes	Imprimez le nombre d'octets
-L, --Max-Line-Longueur	Imprimez la longueur de la ligne la plus longue

Sur mon OS je n'ai pas de fichier APT alors compte du nombre de lignes de `.zshrc` à l'aide de la commande **`wc -l .zshrc`**

Pour moi c'était 108 lignes.

Je peux créer un fichier nommé `nb_lignes.txt` dans lequel je verserais le contenu du fichier source à l'aide de la commande **`wc -l .zshrc > nb_lignes.txt wc -l`**. Le chevron étant là pour rediriger l'output.

Pour afficher le contenu d'un fichier et l'enregistrer dans un autre fichier, il faut faire la commande suivante **`cat .zshrc | tee save_sources`**

On a bien vérifié que le `.zshrc` qui s'affiche dans le terminal tout en ayant une sauvegarde de ce résultat dans un nouveau fichier `save_sources`. Le `cat` permet d'afficher le résultat du fichier `.zshrc`. Le pipe permet de passer le résultat du shell à une autre commande. La commande `tee` vient remplacer le contenu du fichier par le résultat de la commande initiale.

Si l'on souhaite trouver un fichier qui commence par une lettre précise : **`ls -d a*`** si c'est pour un fichier qui va commencer par un point on écrira **`ls -d .*`**

Admettons que je veux afficher tous les fichiers commençant par un "." et contenant le mot "alias" dans ce cas j'utilise la commande : **`ls -a .* | grep -r alias`**. La même commande est possible sur Linux. La commande `grep` permet de chercher la valeur donnée (chez nous `alias`) en sélectionnant les lignes avec une ou plusieurs occurrences.

Pour aller plus loin

Pour installer la commande `tree`: **`brew install tree`**

Pour lancer la commande en arrière-plan et enregistrer le résultat dans un fichier on utilisera la commande **`tree > tree.save &`**

L'esperluette permet d'effectuer l'action en arrière-plan. On retrouvera ainsi le fichier `tree.save` dans le home de l'utilisateur.

On peut vérifier le fichier en tapant `cat tree.save`.

Pour lister le nombre d'éléments du dossier **`ls | wc -l`**

Pour lancer une update et si l'update réussie lancer une upgrade : *update & upgrade*.
Nous avons en effet maintenu nos alias. Même commande sur Linux selon les aliases utilisés.