

SHELL

Un **shell** est un interpréteur de commandes destiné aux systèmes d'exploitation qui permet d'accéder aux fonctionnalités internes du système d'exploitation. Il se présente sous la forme d'une interface en ligne de commande accessible depuis la console ou un terminal. L'utilisateur lance des commandes sous forme d'une entrée texte exécutée ensuite par le shell.

Job 1

- ❖ Afficher le manuel de commande ls: **man ls**

La commande **man** permet de visionner le manuel d'une commande ou le manuel d'un fichier de configuration. Celle-ci est très utile pour obtenir des informations complètes.

La commande **ls** affiche le contenu d'un répertoire. Si aucune option n'est fourni à la commande **ls**, cette dernière affiche le contenu du répertoire courant.

- ❖ Afficher les fichiers cachés du home de votre utilisateur: **ls -a**

L'option **-a** permet d'afficher tous les fichiers des répertoires, y compris les fichiers commençant par un « . »

- ❖ Afficher les fichiers cachés plus les informations sur les droits sous forme de liste : **ls -l -a**

En plus du nom, l'option **-l** de la commande **ls** affiche le type du fichier, les permissions d'accès, le nombre de liens physiques, le nom du propriétaire et du groupe, la taille en octets, et l'horodatage.

Comment ajouter des options à une commande ?

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

Une commande Linux est un programme ou utilitaire qui s'exécute sur la ligne de commande Linux. La ligne de commande (CLI) est un terminal qui permet à l'utilisateur de taper des instructions texte et de les faire exécuter par le système.

pour ajouter une option à une commande il faut mettre un espace et ajouter (-) ou (--) et ajouter l'option de la commande .

Job 2

Commande pour lire un fichier (.bashrc) : **cat .bashrc**

La commande **cat** tire son nom du mot concaténer (concatenate en anglais). Elle permet de créer, de fusionner ou d'imprimer des fichiers dans l'écran de résultat standard ou vers un autre fichier et bien plus encore

Commande pour afficher les 10 premières lignes : **head .bashrc**

head est une commande qui permet d'afficher les premières lignes de texte d'un fichier ou de l'entrée standard.

Par défaut, **head** affiche les 10 premières lignes . Le nombre de lignes affichées peut être changé avec une option.

commande pour afficher les 10 dernières lignes : **tail.bashrc**

tail est une commande qui permet d'afficher les dernières lignes de texte d'un fichier.

Par défaut, **tail** affiche les 10 dernières lignes du fichier. Le nombre de lignes affichées et l'unité utilisée (lignes, blocs, octets) peuvent être changés avec une option.

commande pour afficher les 20 premières lignes : **head + option -n :**
head -n20 .bashrc

La commande **head** avec l'option **-n** permet d'afficher le nombre de lignes que l'on souhaite voir .

Commande pour afficher les 20 dernières lignes : **tail + option -n :**
tail -n20.bashrc

La commande **tail** avec l'option **-n** permet d'afficher le nombre de lignes que l'on souhaite visionner.

- Job 3

APT pour Advanced Package Tool est un utilitaire présent sur les distributions Linux Debian comme Ubuntu, Mint, Kali, etc

Cette commande permet d'installer, supprimer un logiciel ou encore mettre à jour ce dernier.

Installer le paquet "cmatrix" :

sudo apt install cmatrix

lancer le paquet "cmatrix" :

cmatrix

mettre à jour le paquet:

La commande **apt update** recherche les mises à jour disponibles pour le système et les programmes installés en se basant sur les sources définies dans /etc/apt/source.list. Un fichier d'index est créé pour lister les mises à jour disponibles. Il servira de référence pour l'installation de nouvelles mises à jour.

sudo apt update

Mettre à jour ses différents logiciels :

La commande **upgrade** installe les mises à jour identifiées avec **apt update** sans supprimer les paquets déjà installés.

Si la mise à jour d'un paquet nécessite la suppression d'un paquet installé, la mise à jour n'est pas réalisée.

sudo apt upgrade

Télécharger les logiciels : google

Commande **wget** : wget est un client HTTP, HTTPS et FTP, qui permet de télécharger du contenu d'un serveur Web ou FTP.

La commande wget est un utilitaire de ligne de commande Unix / Linux populaire pour récupérer le contenu du Web. Son utilisation est gratuite et offre un moyen non interactif de télécharger des fichiers à partir du Web.

C'est un logiciel libre dont le nom est dérivé de l'anglais World Wide Web et get.

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

Redémarrer votre machine:

La commande shutdown permet d'arrêter ou redémarrer Linux via le terminal. On peut lui ajouter une option suivie par une heure et un message.

Arrêter la machine normalement: **shutdown**

Redémarrer la machine : **shutdown -r**

Job 4

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

Pour créer un fichier on peut utiliser différentes commandes

La commande **touch** est principalement utilisée pour créer des fichiers vides, et modifier le timestamp (horodatage) de fichiers ou de dossiers.

touch users.txt

nano : **Nano** est un éditeur de texte, en console ou en terminal, très utile pour créer et modifier les fichiers de configuration du système.

nano users.txt

Créer un groupe appelé "Plateformeurs"

La commande **addgroup** crée un nouveau compte de groupe en utilisant les valeurs spécifiées sur la ligne de commande et les valeurs par défaut du système. Le nouveau groupe sera inséré dans les fichiers du système selon les besoins.

sudo addgroup Plateformeurs

useradd crée un nouveau compte d'utilisateur en utilisant les options spécifiées sur la ligne de commande plus les valeurs par défaut spécifiées dans le fichier

Créer un utilisateur appelé "User1"

sudo adduser User1

Créer un utilisateur appelé "User2"

sudo adduser User2

Ajouter "User2" au groupe Plateformeurs

sudo adduser User2 Plateformeurs

Copier votre " user.txt" dans un fichier "droits.txt"

Cat : Au lieu d'afficher le contenu d'un fichier dans la console, on peut rediriger le résultat vers un autre fichier en utilisant l'option **">"**.

cat users.txt > droits.txt

Copier votre "user.txt" dans un fichiers "groupes.txt"

`cat users.txt > groupes.txt`

Changer le propriétaire du fichier "droits.txt" pour mettre user1

Commande **chown**: La **commande chown** permet de changer le propriétaire et/ou le groupe.

La commande **chgrp** permet de changer le groupe.

`sudo chown user1 droits .txt`

Changer les droits du fichier "droits.txt" pour que user2 ai accès seulement en lecture

Les droits d'accès définissent la possession d'un fichier ou d'un répertoire à un utilisateur et à un groupe d'utilisateurs. Ils gèrent aussi quelles actions les utilisateurs ont le droit d'effectuer sur les fichiers (lecture, écriture et exécution), selon qu'ils sont propriétaire du fichier, membre du groupe propriétaire du fichier ou ni l'un ni l'autre. La possession et la gestion des permissions associées s'effectuent individuellement avec chaque fichier.

1. Pour changer les droits pour user 2 il faut d'abord modifier le groupe :

`chgrp plateformeurs droits.txt`

2. on modifie ensuite les droits :

`sudo chmod g-w droits.txt`

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

chmod o=r

La commande **chmod** permet de modifier les droits et permissions. On utilise cette dernière en indiquant l'utilisateur (u, g, o ou a), les actions "+, -, =" et le droit concerné "r, w, x". On pourra éventuellement utiliser la notation décimale.

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

1. **sudo chgrp plateformeurs groupes.txt**
2. **chmod g=rw groupes.txt**

Job 5

Un **alias** est une commande personnalisée créée par un utilisateur pour lancer une autre commande.

La plupart du temps, on l'utilise pour exécuter une commande avec beaucoup de paramètres.

Cela fait gagner du temps car il n'est pas nécessaire de ressaisir la commande au complet.

Attention, les alias ajoutés par la commande alias ne sont pas persistants. Vous les perdez une fois déconnecté de votre terminal.

Pour rendre un alias persistant, c'est -à -dire que l'alias reste actif même si on déconnecte notre Shell, il faut éditer le fichier bashrc.

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"
```

Editer les alias dans le bash:

```
sudo nano ~/.bashrc
```

Ajouter une variable d'environnement qui se nommera "USER" et qui sera égale à votre nom d'utilisateur

Les variables d'environnement sont des variables qui sont définies pour le shell en cours d'utilisation et héritées par tous les shells ou processus enfant. Les variables d'environnement servent à transmettre des informations dans les processus qui se déclenchent depuis le shell.

Une variable de shell peut être exportée pour devenir une variable d'environnement grâce à la commande **export**

```
export USER=julie
```

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

La commande **source** peut être utilisée pour recharger/rafraîchir un fichier de fonctions dans le script shell courant ou d'une invite de commande.

source .bashrc

ou redémarrer le terminal .

Afficher les variables d'environnement

La commande **printenv** permet d'imprimer les valeurs de variables d'environnement sur le terminal.

printenv

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

sudo nano .bashrc : editer le fichier : export

PATH=\$PATH:/home/julie/Bureau

puis mettre à jour la modification: **source .bashrc**

Vous devez télécharger l'archive suivante et la désarchiver seulement avec le terminal.

Cette manipulation vous permettra d'accéder à la suite du sujet.

<https://drive.google.com/file/d/11dSelXQuH4tih6zesbv-6OMEpr-sT77X/view?usp=sharing>

wget

<https://drive.google.com/file/d/11dSelXQuH4tih6zesbv-6OMEpr-sT77X/view?usp=sharing>

Le lien n'étant pas utilisable il faut télécharger le fichier manuellement .

extraire une archive tar.gz

`tar xvzf '/home/julie/Téléchargements/Copie de Ghost in the Shell.tar.gz'`

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

La commande `echo` permet d'afficher un texte dans le terminal.

En utilisant la redirection, il est possible de créer un fichier (ou écraser un existant) avec le contenu souhaité.

`echo "je suis votre fichier" > une_commande.txt`

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

La commande `wc` (word count) permet de connaître le nombre de lignes, de mots, d'octets et de caractères dans des fichiers spécifiés par les arguments de fichier.

Pour compter seulement les lignes il faut utiliser l'option `-l` de la commande.

`wc -l /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 /etc/apt/sources.list > save_sources.txt
```

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

Grep est un acronyme qui signifie Global Regular Expression Print. C'est un outil en ligne de commande utilisé pour rechercher une chaîne de caractères dans un fichier spécifié.

find est une commande permettant de chercher des fichiers dans un ou plusieurs répertoires selon des critères définis par l'utilisateur.

Sur tous les fichiers trouvés par find, je veux appliquer une commande, on va utiliser l'option -exec.

- cette commande ne doit pas être entre guillemets, accolades ou \$() ;
- le nom du fichier trouvé est identifié par {}
- la fin de la commande doit être \;

```
find -type f -name "." -print -exec grep "alias" {} \;
```

Pour aller plus loin

Le **&** sert à lancer une commande en arrière plan. Ça veut dire que le shell lance la commande et n'attend pas qu'elle se termine. Le **&** est interprété par le shell.

Un "pipe"|" ou "tuyau" permet de passer le résultat d'une commande shell à une autre commande shell.

L'opérateur "&&" permet de lancer une commande si et seulement si la première (celle à gauche de l'opérateur) s'est correctement terminée.

Le double "||" C'est simplement l'inverse de "&&". La commande à droite de l'opérateur est exécutée seulement si le code de retour de la première commande est différent de 0 (ko).

Installer la commande tree

La commande Tree est similaire à la commande ls qui est connue pour répertorier les fichiers et les répertoires. La seule différence est que la commande tree affiche les fichiers et les répertoires de manière arborescente. Cette commande n'est pas installée par défaut.

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 > save.tree

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

ls -la | 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

update && upgrade