

SHELL

BASE DE CONNAISSANCES

- Interface système
- le Shell, qu'est-ce que c'est ?
- L'histoire de Bash et son utilité ?
- Commande Man
- Commande ls
- Commande apt
- Exemple options shell
- Gestion des droits sur linux
- Les opérateurs du shell
- Les différents opérateurs shell

Shell

Pour pouvoir interpréter et exécuter les commandes qu'on fournit au terminal, le système d'exploitation a besoin d'un autre programme : **un shell**.

D'habitude, on décrit un système d'exploitation Unix à l'aide de la métaphore de la noix. La couche de bas niveau du système s'appelle le noyau (« kernel » en anglais). C'est la partie comestible, et donc utile, de la noix. Elle est protégée par une couche de haut niveau, la coque (« shell » en anglais) de la noix. Pour accéder au noyau qui contient la partie utile du système, il faut obligatoirement passer par la coque.

Le **shell**, c'est donc le logiciel qui fournit l'**interface utilisateur du système d'exploitation**. Ce logiciel peut prendre deux formes. CLI et GUI. En ligne de commande, c'est donc le shell qui interprète et traite les commandes saisies dans le terminal et qui lui fournit un résultat à imprimer. La grande majorité des distributions Unix ont **Bash** comme shell par défaut. Il y a cependant beaucoup d'autres shells, comme par exemple sh, zsh, ksh, tsch, ash, dash et fish. Certains ont des différences en termes de syntaxes, d'autres ont la même syntaxe mais des caractéristiques et fonctionnalités en plus ou en moins.

Source Codequoi.com

- Un "shell" est un programme qui permet à l'utilisateur de communiquer avec le système d'exploitation en utilisant des commandes texte.
- "**Bash**" est l'un des shells les plus populaires sous Unix et Linux, largement utilisé pour ses fonctionnalités avancées et sa compatibilité avec le shell "Bourne" original (d'où son nom "Bourne Again Shell").

SHELL / Ligne de commande de base

- ls : permet de lister le contenu du répertoire
- cd : permet de changer de répertoire
- cp : permet de copier un fichier
- rm : permet de supprimer un fichier
- mv : permet de déplacer un fichier

Commande Man: Sous Linux, qu'est-ce que la commande man ? Disponible **nativement** sous Linux, 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 de l'aide à partir de la ligne de commande. Pour consulter l'aide d'une commande à partir de la commande man elle-même, il suffit d'indiquer le nom de la commande.

man <nom de la commande>. Par exemple, pour accéder au manuel de la commande "**tail**", entrez la commande suivante : **man tail**

Commande Ls:ls écrit dans la sortie standard le contenu de chaque paramètre

Directory spécifié ou le nom de chaque paramètre *File* spécifié, ainsi que toute autre information que vous demandez avec les indicateurs. Si vous ne spécifiez pas de paramètre *Fichier* ou *Répertoire* , la commande ls affiche le contenu du répertoire en cours.

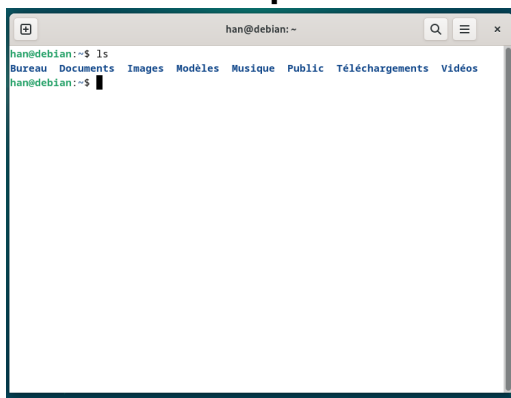
Exercice 1: **Afficher la commande du manuel ls**

La syntaxe de la commande ls :

ls [OPTIONS] [FICHIERS]

pwd : localiser =print working directory

ls : liste le Répertoire = list



Le répertoire est composé de

Bureau/Documents/Images/Musique/Public/Téléchargements/Vidéos

Man, est la commande de manuel. Écrire dans le terminal **man ls**

On obtient ce résultat ci-dessous:

```
han@debian: ~  
LS(1) Commandes de l'utilisateur LS(1)  
  
NOM  
ls - Afficher le contenu de répertoires  
  
SYNOPSIS  
ls [OPTION]... [FICHIER]...  
  
DESCRIPTION  
Afficher les informations des FICHIERs (du répertoire courant par défaut). Les entrées sont triées alphabétiquement si aucune des options -cftuvSUX ou --sort n'est indiquée.  
  
Les paramètres obligatoires pour les options de forme longue le sont aussi pour les options de forme courte.  
  
-a, --all  
inclure les entrées débutant par « . »  
  
-A, --almost-all  
omettre les fichiers « . » et « .. »  
  
--author  
Manual page ls(1) line 1 (press h for help or q to quit)
```

- Afficher les fichiers cachés du home de votre utilisateur

Écrire dans le terminal `man ls -a`. Pour afficher tous les éléments, y compris les éléments cachés, il suffit d'ajouter l'argument -a («all» en anglais): `ls -a`. La commande "`ls`" avec l'option "-a" permet d'indiquer "*all*", c'est-à-dire tous les éléments, y compris ceux qui sont masqués. Il est à noter que l'on peut utiliser également l'option "-A". Pour afficher aussi les fichiers "cachés" (en fait ceux qui commencent par un point) utilisez l'option -A (pour All) :

```
han@debian: ~$ pwd  
/home/han  
han@debian: ~$ ls  
Bureau Documents Images Modèles Musique Public Téléchargements Vidéos  
han@debian: ~$ ls -a  
.  
..  
..bashrc Documents .lessht .profile  
.. Bureau .face .local Public  
..bash_history .cache .face.icon Modèles Téléchargements  
..bash_logout .config Images Musique Vidéos  
han@debian: ~$
```

- Afficher les fichiers cachés plus les informations sur les droits sous forme de liste:

On utilise la commande `ls -la` dans ce cas présent. On obtient une liste complète de tous les fichiers et répertoires, y compris les fichiers cachés, avec des informations détaillées sur les droits et d'autres attributs. Cette commande est couramment utilisée pour inspecter le contenu d'un répertoire de manière exhaustive. Écrire la commande **ls -la**

```
han@debian:~$ ls -al
total 80
drwx----- 13 han  han  4096 27 sept. 11:55 .
drwxr-xr-x  4 root root  4096 25 sept. 12:52 ..
-rw-----  1 han  han    5 25 sept. 14:27 .bash_history
-rw-r--r--  1 han  han  220 25 sept. 12:52 .bash_logout
-rw-r--r--  1 han  han 3526 25 sept. 12:52 .bashrc
drwxr-xr-x  2 han  han  4096 25 sept. 13:46 Bureau
drwx-----  7 han  han  4096 25 sept. 14:29 .cache
drwx-----  9 han  han  4096 25 sept. 14:22 .config
drwxr-xr-x  2 han  han  4096 25 sept. 13:46 Documents
-rw-r--r--  1 han  han 5290 25 sept. 12:52 .face
lrwxrwxrwx  1 han  han    5 25 sept. 12:52 .face.icon -> .face
drwxr-xr-x  3 han  han  4096 27 sept. 11:32 Images
-rw-----  1 han  han   20 27 sept. 11:55 .lessht
drwx-----  4 han  han  4096 25 sept. 13:46 .local
drwxr-xr-x  2 han  han  4096 25 sept. 13:46 Modèles
drwxr-xr-x  2 han  han  4096 25 sept. 13:46 Musique
-rw-r--r--  1 han  han  807 25 sept. 12:52 .profile
drwxr-xr-x  2 han  han  4096 25 sept. 13:46 Public
drwxr-xr-x  2 han  han  4096 25 sept. 13:46 Téléchargements
drwxr-xr-x  2 han  han  4096 25 sept. 13:46 Vidéos
```

- Comment ajouter des options à une commande ?

Pour ajouter des options à une commande sous Debian, vous devez simplement spécifier ces options après le nom de la commande dans le terminal.

- Quelles sont les deux syntaxes principales d'écriture des options pour une commande ? Les options sont généralement précédées d'un tiret court `-` ou d'un double tiret `--`. Par exemple `-a` ou `--all`. Les deux sont équivalents.

Exercice 02 : - Lisez un fichier en utilisant une commande qui permet seulement de lire

- **Linux-CAT** : afficher le fichier entier. L'affichage de gros fichiers submerge la ligne de commande.
- **Linux-LESS** : afficher le fichier entier page par page. Pas très pratique pour les gros fichiers.
- **Linux-HEAD** : afficher le début d'un fichier ou limiter l'affichage à une certaine zone.
- **Linux-TAIL** : afficher la fin d'un fichier ou restreindre l'affichage à une certaine zone.

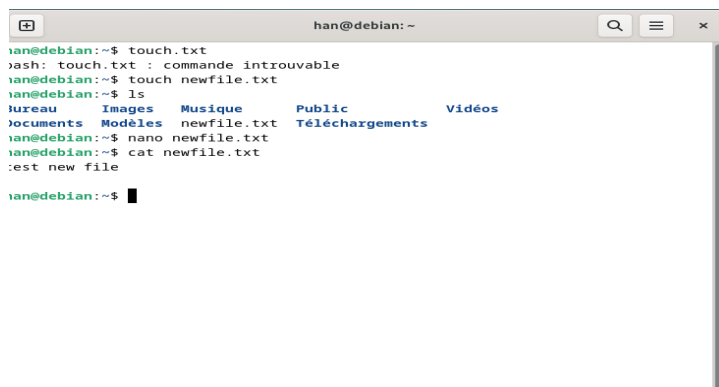
Il faut donc créer un fichier .txt. Ensuite le lire avec la commande **cat**.

Se créer un fichier avec "touch" (nom du fichier.txt)

faire **ls** Le fichier est visible

faire **nano newfile.txt** La commande nano : vous pouvez écrire dans le fichier
texte

faire **cat newfile.txt** La commande cat : vous permet de lire le fichier



```
han@debian: ~  
ian@debian:~$ touch.txt  
ash: touch.txt : commande introuvable  
ian@debian:~$ touch newfile.txt  
ian@debian:~$ ls  
Bureau  Images  Musique  Public  Vidéos  
Documents  Modèles  newfile.txt  Téléchargements  
ian@debian:~$ nano newfile.txt  
ian@debian:~$ cat newfile.txt  
cest new file  
ian@debian:~$
```

Pour afficher les 10 premières lignes du fichier .bashrc

La commande "head" est intégrée à de nombreuses distributions [Linux](#), notamment Debian. Elle permet d'afficher le contenu d'un fichier en commençant par le début (d'où le nom de ce programme, probablement). Ainsi, si l'on appelle la commande head suivie d'un nom de fichier, sans préciser d'options, les 10 premières lignes du fichier s'affichent à l'écran. Ecrire la commande ci-dessous:

head .bashrc

Pour afficher les 10 dernières lignes du fichier .bashrc

La commande tail de Linux est l'un des outils essentiels de l'interface en ligne de commande. La fonction principale de cette commande est d'afficher la fin d'un fichier (texte), autrement dit de limiter la sortie d'une commande Linux à un certain nombre de lignes. La commande tail fait donc partie d'une série qui inclut la commande head de Linux et les commandes 'cat' et 'less'. Les [commandes Linux](#) servent toutes à afficher le contenu de fichiers texte

La commande tail de Linux est appelée dans l'interface en ligne de commande.

Comme il est d'usage, le nom de la commande est saisi, suivi de paramètres facultatifs. La séquence se termine par le nom ou le chemin d'un ou de plusieurs fichiers.

Ecrire la commande ci-dessous:

tail .bashrc

A terminal window titled 'han@debian: ~' with search, menu, and close buttons. The command 'tail .bashrc' has been executed, displaying the contents of the .bashrc file from the last 10 lines. The output shows configuration for bash completion, including comments and conditional logic to source completion scripts from /usr/share or /etc.

```
han@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
han@debian:~$
```

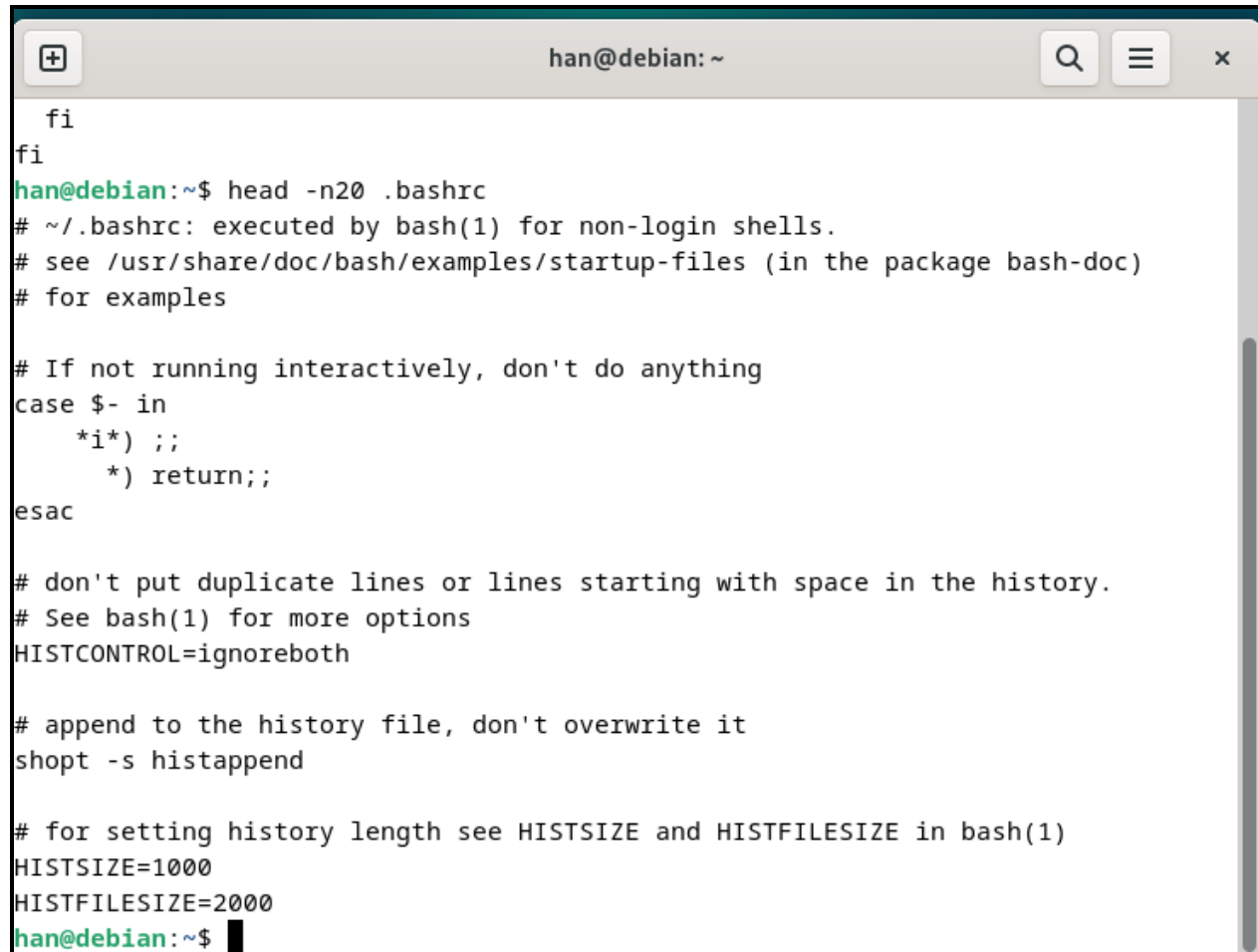
Lorsque vous avez besoin d'imprimer un nombre spécifique de lignes, vous pouvez utiliser l'option -n suivie du nombre de lignes.

Pour afficher les 20 premières lignes du fichier .bashrc

Ecrire la commande ci-dessous:

head [option] [fichier]

head -n 20 .bashrc

A terminal window titled 'han@debian: ~' with search, menu, and close buttons in the title bar. The terminal shows the command 'head -n20 .bashrc' and its output, which is the first 20 lines of the .bashrc file. The output includes comments about non-login shells, interactive shell handling, history control, and history file settings.

```
fi
fi
han@debian:~$ head -n20 .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
han@debian:~$
```

Pour afficher les 20 dernières lignes du fichier .bashrc

Sortir la fin d'un fichier à l'aide de la commande tail de Linux

Dans le cas le plus simple, on utilise la commande tail de Linux pour sortir les dernières lignes d'un fichier. Pour l'essayer, remplacez le caractère générique '<fichier>' par le nom ou le chemin d'un fichier texte présent sur votre système. Ici, nous utilisons l'option -n et nous sortons les trois dernières lignes d'un fichier :

tail [option] [fichier]

tail -n 20 .bashrc


```
han@debian: ~  
HISTSIZE=1000  
HISTFILESIZE=2000  
han@debian:~$ tail -n20 .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  
han@debian:~$
```

Exercice 03 : On nous propose d'utiliser le logiciel **cmatrix** pour simuler **matrix**.
Matrix est un protocole standard ouvert et léger pour la communication en temps réel. [quote Wikipédia](#)

Pour Installer le paquet “cmatrix” :

Ouvrir le terminal

Taper la commande apt pour découvrir son utilisation

Commande Apt: apt“Advanced Package Tool” est un outil en ligne de commande pour gérer les paquets. Il fournit des commandes pour chercher et gérer autant que pour rechercher des informations à propos des paquets. Il fournit les mêmes fonctions que les outils APT spécialisés, tels qu’apt-get et apt-cache, mais dispose d’options plus adaptées pour une utilisation interactive.

La syntaxe d'apt est la suivante :

apt [méthode] [paramètres]

1.1 Méthodes

1.1.1 Méthodes courantes

méthode	paramètre	description
list	nom(s) du ou des paquet(s)	Liste les paquets selon leur nom.
search	texte à rechercher	Liste les paquets dans les descriptions desquels on trouve le texte correspondant.
show	nom(s) du ou des paquet(s)	Affiche les détails du paquet.
install	nom(s) du ou des paquet(s)	Installe les paquets.
remove	nom(s) du ou des paquet(s)	Supprime les paquets.
autoremove	pas de paramètre	Supprime automatiquement les dépendances inutilisées (n'existe pas sous Trusty , utilisez apt-get autoremove à la place)
update	pas de paramètre	Met à jour <i>la liste</i> des paquets disponibles (mais pas les paquets eux-mêmes).
upgrade	pas de paramètre	Met à jour le système en installant/mettant à jour les paquets.
full-upgrade	pas de paramètre	Met à jour le système en supprimant/installant/mettant à jour les paquets.
edit-sources	pas de paramètre ou nom du fichier .list	Permet d'éditer les sources de logiciels .

Certaines méthodes requièrent l'utilisation de la commande [sudo](#), d'autres pas, selon qu'elles influent ou non sur les fichiers du système.

`sudo apt install mon_paquet`

Remplacez « mon_paquet » par le nom de votre paquet. Écrire cette commande:

sudo apt install cmatrix On lance Matrix

cmatrix Voici le programme Matrix

```
^ k a C? 1 5 $B1X p ;%< H f * L ]
3 N > d t U # c g 9&q$ _ iK C = V s
1 a z i z q : U 1 W R e e y h ^ > a - b -
N s w F p @ Y x C + # X z o 7 B _ -
/AQ m 4 9 u p @ ; ) 0 @ ) h U e = 8 !
) 1 n D A % J g G 8 Q a o ] f Z < ) >
M e Z 2 3 Y m [ 8 3 N 7 6 N R 8 T / /
( $ ? u Y L u 2 f > - L . Q ( l 7 6
> W f ^ d i @ s u X F o ? X 0 > 0 :
g k / q t ^ J s 5 R V 2 * 3 T 1 F +
E 2 \ s ! ^ u M o U s t v T N ^ o o
j 7 n x & d b [ Y I F P R p U d X Y y l
(> Z a j 5 j n ? [ " e Y i n 2 W # i t \
H F I B . W E > K " b D 2 i t z > b K > <
x 3 = F i / o 1 N ( I d 2 - ] ) ; d i i Z Y H
f Y 4 M f B U @ t 9 ; [ 3 x \ Y ? 0 g ( s ) 5 h x 9
i 1 - 2 ( ) : 4 d e D C O & V $ N * h B d L F X t h " S
Y X U v N $ w j , G < O N ] u J \ x T 4 Q + 5 _ V u p *
* W O i D i 0 x s t m 5 ' P " h ; ; n e 4 i Z $ _ O t : _
M W F @ f i = W i s q f \ q R n L g : i _ G \ z " d M
z F , t k N C s ] $ k Q ^ # F x ) P v d N : a " R p
C D 6 Y 2 ) h 1 ' * ? M N h , I + + S $ H m 7 N w
; 7 x % I G E 0 z f t / $ 4 \ P H O z H 8 %
```

sudo

On quitte Matrix : CTRL C

Mettre à jour son gestionnaire de paquets:

Commande : **update** -met à jour la liste des paquets disponibles

La commande update permet simplement d'actualiser la liste des paquets disponibles pour votre système.

On lance cette commande avant d'installer un nouveau paquet ou avant d'installer les mises à jour du système.

Écrire cette commande ci-dessous

sudo apt update avec ou sans sudo

```
root@debian:/etc/apt# apt upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@debian:/etc/apt# sudo apt update
Ign :1 cdrom://[Debian GNU/Linux 12.0.0 _Bookworm_ - Official amd64 DVD Binary-1 with f
irmware 20230610-10:23] bookworm InRelease
Err :2 cdrom://[Debian GNU/Linux 12.0.0 _Bookworm_ - Official amd64 DVD Binary-1 with f
irmware 20230610-10:23] bookworm Release
Veuillez utiliser apt-cdrom afin de faire reconnaître ce cédérom par votre APT. apt-g
et update ne peut être employé pour ajouter de nouveaux cédéroms
Atteint :3 http://deb.debian.org/debian bookworm InRelease
Atteint :4 http://deb.debian.org/debian-security bookworm-security InRelease
Atteint :5 http://deb.debian.org/debian bookworm-updates InRelease
Lecture des listes de paquets... Fait
E: Le dépôt cdrom://[Debian GNU/Linux 12.0.0 _Bookworm_ - Official amd64 DVD Binary-1 w
ith firmware 20230610-10:23] bookworm Release n'a pas de fichier Release.
N: Les mises à jour depuis un tel dépôt ne peuvent s'effectuer de manière sécurisée, et
sont donc désactivées par défaut.
N: Voir les pages de manuel d'apt-secure(8) pour la création des dépôts et les détails
de configuration d'un utilisateur.
root@debian:/etc/apt#
```

Mettre à jour ses différents logiciels

Commande : **upgrade** -met à jour le système en installant/mettant à jour les paquets

Mettez à jour tous les paquets installés sur votre système Debian

```
root@debian:/etc/apt# apt upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@debian:/etc/apt#
```

Télécharger les internets : Google

Cette commande utilise wget pour télécharger la page d'accueil de Google. Par défaut, wget enregistre le fichier "index.html" dans le répertoire actuel. C'est à dire avoir une copie d'une page google.com

Ecrire la ligne de commande suivante

wget google.com

```
han@debian:~$ wget google.com
--2023-09-28 12:04:52-- http://google.com/
Résolution de google.com (google.com)... 142.251.37.238, 2a00:1450:4006:80e::200e
Connexion à google.com (google.com)|142.251.37.238|:80... connecté.
requête HTTP transmise, en attente de la réponse... 301 Moved Permanently
Emplacement : http://www.google.com/ [suivant]
--2023-09-28 12:04:52-- http://www.google.com/
Résolution de www.google.com (www.google.com)... 142.250.201.4, 2a00:1450:4006:808::2004
Connexion à www.google.com (www.google.com)|142.250.201.4|:80... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : non indiqué [text/html]
Sauvegarde en : « index.html »

index.html          [ <=> ] 22,26K --.-KB/s   ds 0s

2023-09-28 12:04:52 (108 MB/s) - « index.html » sauvegardé [22790]
```

Redémarrer le terminal

Être dans le root puis écrire la commande **exit**

Éteindre votre machine

Ecrire la commande **Ctrl + D**

Exercice 04:

Certaines lignes de code à apprendre

vm =

less =

cat =

nano = écrire

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

Créer un fichier users.txt

touch users.txt

Ensuite écrire User1 et User2 dans le fichier nommé users.txt. Pour cela aller sur

nano users.txt

Ecrire User1 et User2 dans nano

Quitter, OK, Enter

Créer un groupe appelé "Plateformeurs"

Être dans le root

Ecrire la commande **sudo groupadd "Plateformeurs"**

Créer un utilisateur appelé "User1"

Mode root su

commande : **useradd User1**

Créer un utilisateur appelé "User2"

Mode root su

commande : **useradd User2**

Ajouter "User2" au groupe Plateformeurs

La commande **gpasswd** permet d'ajouter ou de supprimer un utilisateur d'un groupe. Avec l'option **--add**, gpasswd ajoute un utilisateur à un groupe.

```
root@debian~#: gpasswd --add nom-utilisateur nom-groupe
```

```
/
root@debian:/# sudo useradd User2
root@debian:/# gpasswd User2 Plateformeurs
Utilisation : gpasswd [option] GROUP

Options :
-a, --add USER          ajouter USER à GROUP
-d, --delete USER       supprimer USER de GROUP
-h, --help               afficher ce message d'aide et quitter
-Q, --root CHROOT_DIR   répertoire dans lequel faire le chroot
-r, --remove-password    supprimer le mot de passe de GROUP
-R, --restrict           restreindre l'accès à GROUP à ses membres
-M, --members USER,...  définir la liste des membres de GROUP
-A, --administrators ADMIN,... définir la liste des administrateurs de GROUP

À part les options -A et -M, les options ne peuvent pas être combinées.
root@debian:/#
```

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

En premier créer un fichier users.txt et un fichier droits.txt

commande **touch users.txt** **touch droits.txt**

Pour copier : La commande cp :

cp [Original] [Destination]

sudo cp users.txt "droits.txt"

Pour supprimer :

La commande rm permet d'effacer un fichier et tout son contenu.

rm nom_du_fichier

Pour déplacer :

La commande mv (pour move) permet de [déplacer un fichier, répertoire ou toute arborescence sur Linux](#). Elle s'utilise donc lorsque vous désirez déplacer un fichier ou répertoire d'un emplacement vers un autre. Mais on peut aussi l'utiliser pour renommer un fichier. **Ainsi, c'est une [des commandes de base Linux](#) à connaître au côté de [cp](#), [rm](#) ou [ls](#) pour manipuler les fichiers.**

Copier votre "users.txt" dans un fichier "groupes.txt"

```
sudo cp users.txt groupes.txt
```

Changer les droits:

Changer le propriétaire du fichier "droits.txt" pour mettre "User1"

Pour changer de propriétaire, utiliser la commande chown (change owner)

```
chown User1 droits.txt
```

Changer les droits du fichier "droits.txt" pour que "User2" ai accès seulement en lecture

```
ls -l droits.txt            voir les droits du fichier
```

Voici le résultat -rw-r- - r- - 1 User1 han 0 29 sept. 09:08 droits.txt

Suivent ensuite 3 groupes de 3 symboles chacun, indiquant si le fichier (ou répertoire) est autorisé en lecture, écriture ou exécution. Les 3 groupes correspondent, dans cet ordre, aux droits du propriétaire, du groupe puis du reste des utilisateurs. Dans le paragraphe introductif, vous aurez remarqué des lettres en gras dans les termes anglais. Ce sont ces lettres qui sont utilisées pour symboliser les dites permissions. Si la permission n'est pas accordée, la lettre en question est remplacé par « - ». Si l'on reprend les lettres données pour lecture/écriture/exécution (read/write/execute), nous obtenons : **rwX**.

*Seuls root et le propriétaire d'un fichier peuvent changer ses permissions d'accès. Pour cela, on peut utiliser la commande **chmod**. La commande "chmod" permet de définir qui peut lire, écrire ou exécuter un fichier ou un répertoire.*

```
chmod o+r droits.txt
```

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

```
chmod a+r groupes.txt
```

a (all, tous) représente l'ensemble des trois catégories.

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

g (group, groupe) représente la catégorie "groupe propriétaire"

+ : ajouter

r : read ⇒ lecture

w : write ⇒ écriture

```
chmod g+rw droits.txt
```

Job 5: Votre fichier de documentation contiendra les actions ci-dessous ainsi que leur équivalent en ligne de commande Linux :

LES ALIAS:

Il existe la possibilité de créer des **alias** de commandes qui sont "*des raccourcis*" vers des commandes plus complexes et/ou plus longues. Autrement dit, l'alias est un nom alternatif ou un nom court pour une autre commande. Les alias écrivent sur

Ajouter un alias qui permettra de lancer la commande "ls -la" en tapant "la"

Pour créer un nouvel alias, on utilise la commande alias :

alias [NomAlias]='[commande avec ses options ou arguments]'

```
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'
```

LES VARIABLES

Une variable définition: Une variable est un emplacement mémoire utilisé dans un programme ou un script pour conserver une valeur et pouvoir l'utiliser par la suite.

Un environnement: Au cours d'une interaction avec votre serveur via une session shell, shell compile de nombreuses informations pour déterminer son comportement et accéder aux **ressources**. Certains de ces réglages se font dans les **paramètres de configuration**, d'autres doivent être saisis par l'**utilisateur**. Le shell assure notamment le suivi de tous ces **paramètres** et ces détails par le biais d'une zone qu'il gère, appelée **environnement**. L'environnement est une zone que le shell construit à chaque fois qu'il démarre une session qui contient des variables définissant les propriétés du système.

Une variable d'environnement définition: 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.

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

Un fichier BASHRC est un script Linux qui s'exécute automatiquement chaque fois qu'un utilisateur se connecte ou ouvre un nouveau terminal. Ce script est utilisé pour configurer l'environnement de ligne de commande, définir des variables d'environnement et effectuer d'autres tâches courantes.

Le caractère **\$** permet d'obtenir la valeur d'une variable.

cat .bashrc On peut voir le contenu du dossier .bashrc

On écrit ensuite dans le dossier .bashrc en utilisant la commande nano (logiciel de texte)

nano .bashrc

On se retrouve dans .bashrc

```
#!/usr/bin/env bash
# ~/.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

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement
^X Quitter   ^R Lire fich.^Y Remplacer  ^U Coller    ^J Justifier ^_ Aller ligne
```

Ecrire une variable : nom_de_la_variable=valeur_de_la_variable.

Dans cet exercice on ajoute une variable d'environnement et elle qui se nomme "USER" et elle est égale à votre nom d'utilisateur. Donc on écrit dans le dossier caché .bashrc la catcommande suivante:

han=USER

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

Pour mettre à jour les modifications du fichier caché .bashrc dans le shell, il faut écrire la commande **\$ source .bashrc**

Une méthode pour vérifier, on peut exécuter la commande echo \$.

```
han@debian:~$ echo $USER
han
han@debian:~$
```

Afficher les variables d'environnement

env

```
6:*.*midi=00;36:*.*mka=00;36:*.*mp3=00;36:*.*mpc=00;36:*.*ogg=00;36:*.*ra=00;36:*.*wav=00;36:*.*oga=00;36:*.*opus=00;36:*.*spx=00;36:*.*xspf=00;36:*.*~=00;90:*.*#=00;90:*.*bak=00;90:*.*old=00;90:*.*orig=00;90:*.*part=00;90:*.*rej=00;90:*.*swp=00;90:*.*tmp=00;90:*.*dpkg-dist=00;90:*.*dpkg-old=00;90:*.*ucf-dist=00;90:*.*ucf-new=00;90:*.*ucf-old=00;90:*.*rpmnew=00;90:*.*rpmorig=00;90:*.*rpmsave=00;90:
XDG_CURRENT_DESKTOP=GNOME
VTE_VERSION=7006
WAYLAND_DISPLAY=wayland-0
GNOME_TERMINAL_SCREEN=/org/gnome/Terminal/screen/79bca161_8776_41cc_8bac_d5d4568f9a80
GNOME_SETUP_DISPLAY=:1
XDG_SESSION_CLASS=user
TERM=xterm-256color
USER=han
GNOME_TERMINAL_SERVICE=:1.87
DISPLAY=:0
SHLVL=1
QT_IM_MODULE=ibus
XDG_RUNTIME_DIR=/run/user/1000
XDG_DATA_DIRS=/usr/share/gnome:/usr/local/share:/usr/share/
PATH=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
GDMSESSION=gnome
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
_=/usr/bin/env
han@debian:~$
```

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

[La variable d'environnement](#) \$PATH est une liste de répertoires délimitée par deux points qui indique à l'interpréteur de commandes les répertoires dans lesquels il doit rechercher les fichiers exécutables.

```
export PATH=$PATH:/home/han/Bureau
```

Pour vérifier, on utilise la commande source \$PATH, on obtient bien le chemin souhaité:

```
han@debian:~$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games:/home/han/Bureau
han@debian:~$ s █
```

Job 6 : **Vous devez télécharger l'archive suivante et la désarchiver seulement avec le terminal. Cette manipulation vous permettra d'accéder à la suite du sujet.**

Je suis allé.e dans ma vm Debian 12, j'ai ensuite téléchargé via l'email le fichier "Ghost in the Shell.tar". Pour cela, je suis allé dans le Terminal. J'ai localisé Téléchargements. J'ai écrit la ligne de commande **cd /home/han/Téléchargements** ensuite j'ai tapé la ligne de code **tar -xvf "Ghost in the Shell"**

Le fichier pdf est accessible depuis mes Téléchargements dans la Debian.

Job 7: Les caractères > < >>

> L'opérateur > en Bash est utilisé pour rediriger la sortie standard (stdout) d'une commande vers un fichier

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

```
han@debian:~$ wc -l "une commande.txt"
1 une commande.txt
han@debian:~$
```

J'utilise la commande **wc** words count . **wc -l "une commande.txt"**

Résultat: **1 une commande.txt**

La commande **wc** (qui signifie "word count" ou "compteur de mots" en anglais) est une commande utilisée dans les systèmes Unix et Unix-like, y compris Linux et macOS, pour compter les mots, les caractères et les lignes dans un fichier ou une entrée de texte. Elle est polyvalente et fournit plusieurs informations sur un fichier ou une entrée de texte donnée. Les principales options couramment utilisées de la commande **wc** sont les suivantes :

- `-l` : Compte les lignes.
- `-w` : Compte les mots.
- `-c` : Compte les caractères.

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

cat "une commande.txt">"save_sources"

cat permet d'afficher et > de déplacer le dossier dans save_sources ?

```
han@debian:~$ ls
Bureau      groupes.txt  Modèles    save_sources  users.txt
Documents   Images      Musique    Téléchargements  Vidéos
droits.txt  index.html  Public     'une commande.txt'
```

han@debian:~\$ █

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

On peut peut-être utiliser l'option find.

Suite

Installer la commande 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"

Lister les éléments présents dans le dossier courant est utilisé directement le résultat de votre première commande pour compter le nombre d'éléments trouvés

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

Je ne l'ai pas fais.