DOSSIER LINUX

Contenu

[I. Système d’exploitation 3](#_Toc32936961)

[A. Linux 3](#_Toc32936962)

[1. Architecture 3](#_Toc32936963)

[2. La console 4](#_Toc32936964)

[3. Connexion via SSH 19](#_Toc32936965)

[4. compiler un programme depuis les sources 32](#_Toc32936966)

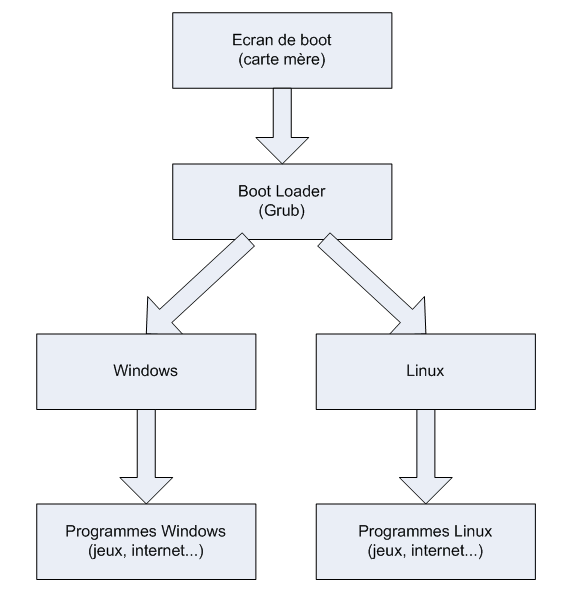
[5. Les scripts bach 33](#_Toc32936967)

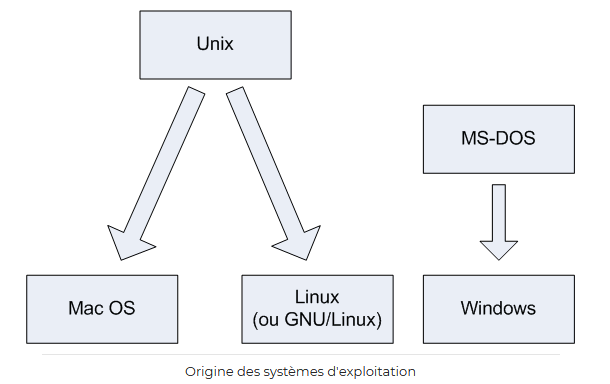
# Système d’exploitation

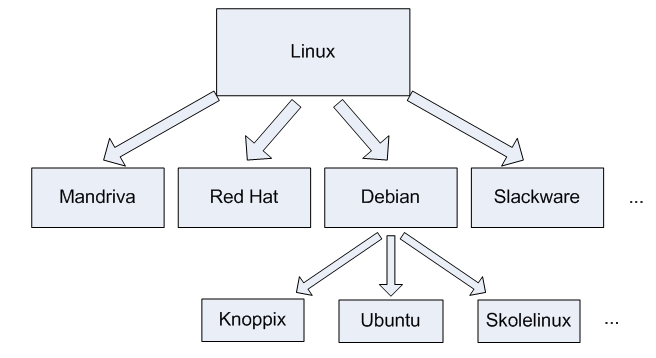
## Linux

### Architecture

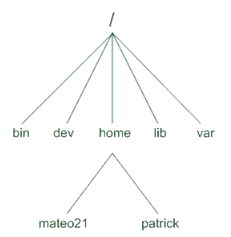
Richard Stallman créa le projet GNU







### 



Racine :

* **bin** : contient des programmes (exécutables) susceptibles d'être utilisés par tous les utilisateurs de la machine.
* **boot** : fichiers permettant le démarrage de Linux.
* **dev** : fichiers contenant les périphériques. En fait – on en reparlera plus tard – ce dossier contient des sous-dossiers qui « représentent » chacun un périphérique. On y retrouve ainsi par exemple le fichier qui représente le lecteur CD.
* **etc** : fichiers de configuration.
* **home** : répertoires personnels des utilisateurs. On en a déjà parlé un peu avant : c'est dans ce dossier que vous placerez vos fichiers personnels, à la manière du dossierMes documentsde Windows.
* **lib** : dossier contenant les bibliothèques partagées (généralement des fichiers.so) utilisées par les programmes. C'est en fait là qu'on trouve l'équivalent des.dllde Windows.
* **media** : lorsqu'un périphérique amovible (comme une carte mémoire SD ou une clé USB) est inséré dans votre ordinateur, Linux vous permet d'y accéder à partir d'un sous-dossier demedia. On parle de **montage**.
* **mnt** : c'est un peu pareil quemedia, mais pour un usage plus temporaire.
* **opt** : répertoire utilisé pour les add-ons de programmes.
* **proc** : contient des informations système.
* **root** : c'est le dossier personnel de l'utilisateur « root ». Normalement, les dossiers personnels sont placés dans home, mais celui de « root » fait exception. En effet, comme je vous l'ai dit dans le chapitre précédent, « root » est le superutilisateur, le « chef » de la machine en quelque sorte. Il a droit à un espace spécial.
* **sbin** : contient des programmes système importants.
* **tmp** : dossier temporaire utilisé par les programmes pour stocker des fichiers.
* **usr** : c'est un des plus gros dossiers, dans lequel vont s'installer la plupart des programmes demandés par l'utilisateur.
* **var** : ce dossier contient des données « variables », souvent des logs (traces écrites de ce qui s'est passé récemment sur l'ordinateur).

### La console

Pour ouvrir la console en mode graphique CTRL + T, ou ATRL +ALT +F1 a F7. Toute commande est décris dans le manuel en tapant dans la console : man + commande

***$Date***

**Description :** Donne la date

**Paramètre :**

***$ls***

**Description :** liste les fichiers et dossier

**Paramètre :**

1. **–a (--all) :** affiche même les fichiers cachés.
2. **–F :** indique le type d’élément
3. **–l :** liste détaillé.
4. **–h :** Affiche la taille en Ko, Mo, Go.
5. **–t :** trie par date de dernière modification.

***$pwd***

**Description :** Affiche le dossier actuel

**$which**

**Description :** Affiche l’emplacement d’une commande

**$cd**

**Description :** change de dossier.

**$du**

**Description :** taille occupée par les dossiers

**Paramètre :**

1. **–h :** Affiche la taille en Ko, Mo, Go.
2. **–a :** Affiche la taille des dossiers et des fichiers.
3. **–s :** Affiche juste la taille total.

**$cat**

**Description :** affiche tout le fichier

**$less**

**Description :** affiche le fichier page par page

**$head**

**Description :** affiche le début d’un fichier

**Paramètre :**

1. **–n + nombre:** affiche les N premières lignes

**$tail**

**Description :** affiche la fin d’un fichier

**Paramètre :**

1. **–n + nombre:** affiche les N dernières lignes
2. **–f :** Permet de suivre un fichier en cours d’écriture.

**$touch**

**Description :** crée un fichier.

**$mkdir**

**Description :** crée un dossier

**Paramètre :**

1. **–p :** Crée tous les dossiers intermédiaires ex : mkdir -p animaux/vertebres/chat

**$cp**

**Description :** copie un fichier le 1er est le fichier copié le deuxième le fichier crée

**Paramètre :**

1. **–R :** Copie les dossiers ainsi que les sous-dossiers.

**$mv**

**Description :** déplace un fichier le 1er est le fichier à déplacer le deuxième le chemin

**$rm**

**Description :** supprime des fichiers et dossiers

**Paramètre :**

1. **–i :** Demande confirmation
2. **–f :** Force la suppression quoi qu’il arrive
3. **–v :** décris ce qu’il se fait.
4. **–r :** Supprime un dossier et tout son contenu

**$ln**

**Description :** crée un lien entre deux fichiers (par défaut ; physiques)

**Paramètre :**

1. **–s :** Crée un lien symbolique

**$sudo**

**Description :** Exécute une commande en root.

**Paramètre :**

1. **su:** Devenir root

**$adduser**

**Description :** ajouter un utilisateur

**$passwd**

**Description :** changer le mot de passe

**$deluser**

**Description :** supprimer un compte

**$addgroup**

**Description :** Ajoute un groupe.

**$** **usermod**

**Description :** modifie un utilisateur.

**Paramètre :**

1. **–l :** renomme l’utilisateur
2. **–g :** change de groupe

**$delgroup**

**Description :** supprime un groupe

**$chown**

**Description :** Gère les propriétaires d’un fichier

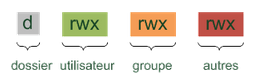
**Paramètre :**

1. **propriétaire : groupe :** Change le propriétaire d’un fichier ainsi que son groupe propriétaire.
2. **- R :** Affecte les sous dossiers

**$chgrp**

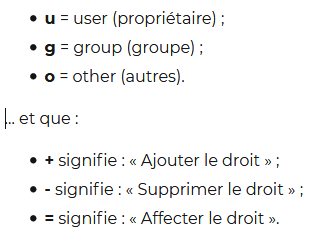
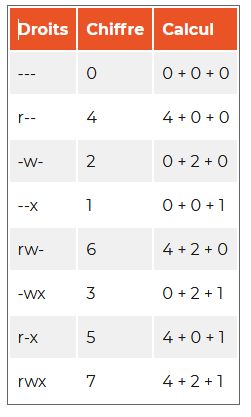
**Description :** Change le groupe propriétaire d’un fichier

**$chmod**

**Description :** modifie les droits d’accès.

**Paramètre :**

1. **- R :** Affecte les sous dossiers



**$apt-get + update**

**Description :** mettre à jour le cache des paquets

**$apt-get + search**

**Description :** rechercher un paquet

**$apt-get + install**

**Description :** télécharge et installe un paquet

**$apt-get + upgrade**

**Description :** Va mettre à jour l’intégralité des paquets en allant chercher sur internet , utiliser **$apt-get + update** avant afin de mettre à jour le cache.

**$apt-get + remove**

**Description :** désinstalle un paquet

**$man + commande**

**Description :** utilise le manuel afin d’avoir une description détaillé de la commande.

Résumé de la syntaxe du SYNOPSIS

Voici un petit résumé de la syntaxe du SYNOPSIS pour vous souvenir de la façon dont chaque élément doit être interprété :

* gras : tapez le mot exactement comme indiqué ;
* souligne : remplacez le mot souligné par la valeur qui convient dans votre cas ;
* [-hvc] : toutes les options -h, -v et -c sont facultatives ;
* a|b : vous pouvez écrire l'option « a » OU « b », mais pas les deux à la fois ;
* option… : les points de suspension indiquent que l'option peut être répétée autant de fois que vous voulez.

**$apropos + mots clé**

**Description :** cherche une commande en fonction d’un mot clés.

**$commande -h**

**Description :** utilise l’aide décrivant la commande.

**$whatis + commande**

**Description :** une petite description de la commande

**$locate + fichier**

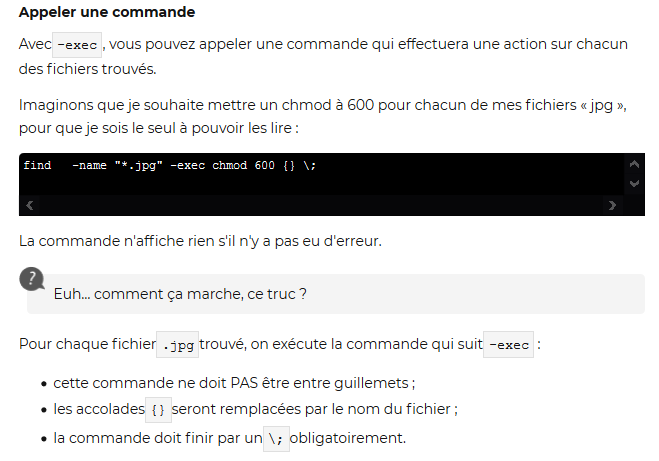
**Description :** va rechercher un fichier dans une base de donnée. Son défaut est si la base de donnée n’est pas à jour elle ne pourra pas trouver le fichier, cas lorsque vous venez de créer un fichier. ( utiliser la commande **$sudo updatedb** )

**$find**

**Description :** cherche un fichier dans le dossier et sous dossier par défaut, un répertoire peut être décris et exécute ce qu’on lui demande.

**Paramètre :**

1. **-name :** cherche un fichier via son nom « nom ».
2. **-size (-)(+) :** cherche un fichier via sa taille k=kilos, M=Méga et G = Giga
3. **-atime :** cherche un fichier depuis la date de dernière modification commence par 0 pour un jour.
4. **-type (d)(f) :** cherche un fichier via son type d = directories, et f= files
5. **-printf  :** voir manuel, peut afficher les résultats en fonction de ce que l’on veut.
6. **-delete :** supprime les fichiers trouvés
7. **-exec :** voir ci-dessous.

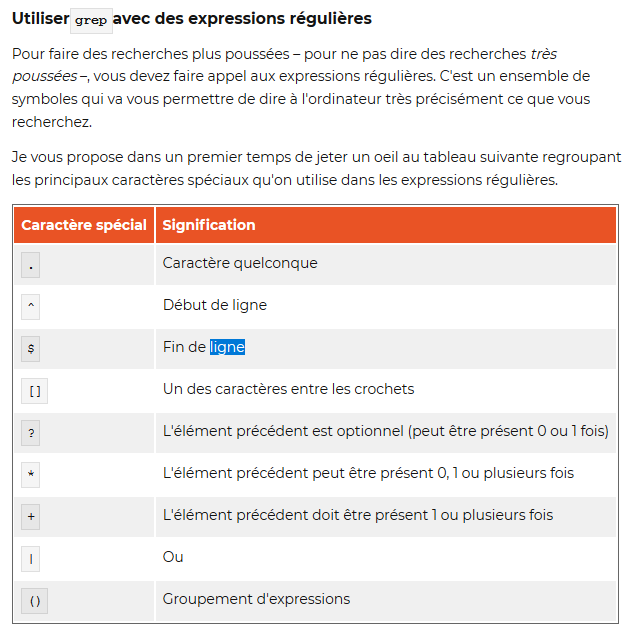


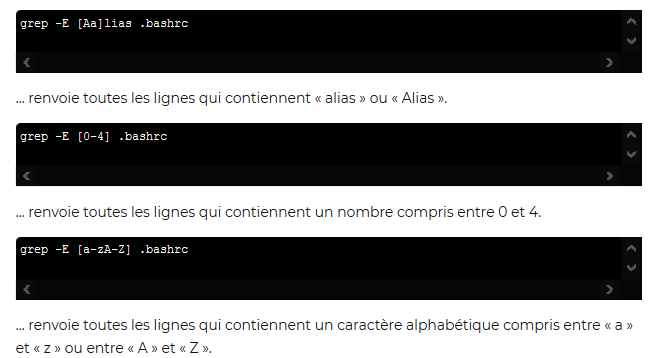
**$grep+filtre+fichier**

**Description :** filtre les données d’un fichier utilisé des guillemets lorsque le filtre est une suite de plusieurs mots.

**Paramètre :**

1. **-i :** ne tiens pas compte de la CASSE
2. **-n :** connaître le numéro des lignes.
3. **-v :** inverse la recherche ignore un mot.
4. **-r :** recherche dans tout les fichiers et sous dossiers, **$grep+filtre+dossier/**
5. **-E :** Utilise des expressions régulière voir ci-dessous.





**$** **sort + fichier**

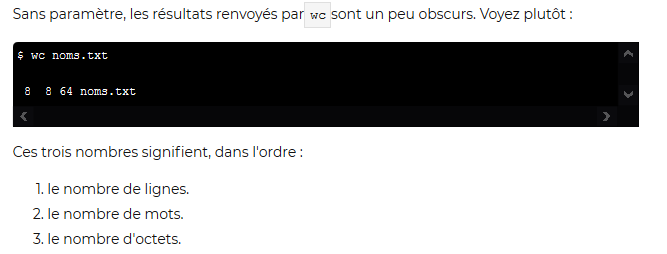
**Description :** trie des lignes

**Paramètre :**

1. **–o :** écrit le résultat dans un fichier **$sort + fichiertrié +fichierenregistré**
2. **–r :** trie en ordre inverse
3. **–R :** trie de manière aléatoire
4. **–n :** trie des nombres

**$** **wc + fichier**

**Description :** compte le nombre de ligne.



**Paramètre :**

1. **–l :** compte le nombre de ligne
2. **–w :** compte le nombre de mots
3. **–c :** compte le nombre d’octets
4. **–m :** compte le nombre de caractère

**$** **uniq + fichier**

**Description :** supprime les doublons ( peut être enregistré dans un fichier **$uniq fichier1 fichier2** )

**Paramètre :**

1. **–c :** compte le nombre d’occurrences
2. **–d :** affiche uniquement les lignes présentes en double

**$** **cut + fichier**

**Description :** coupe une partie du fichier



**Paramètre :**

1. **–d :** indique le délimiteur
2. **–f :** indique le numéro du ou des champs à couper



**$** (chevron) Commande + **>** fichier

**Description :** redirige le résultat dans un fichier, sera créer si il n’existe pas, ou supprime les données si il y en avait déjà.



**$** (double chevron) Commande + >**>** fichier

**Description :** redirige le résultat à la fin d’un fichier, sera créer si il n’existe pas.



**$** (2 chevron) Commande + >>fichier **2>** fichier

**Description :** redirige les erreurs dans un fichier. (2>> peut être utiliser pour ajouter à la fin d’un fichier )



**$**Commande + >>fichier **2>&1**

**Description :** redirige les erreurs le même fichier standard.



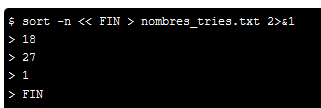
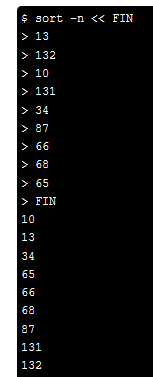
**$**commande **<** fichier

**Description :** lit depuis un fichier



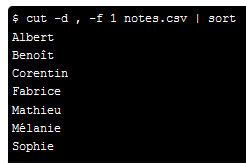
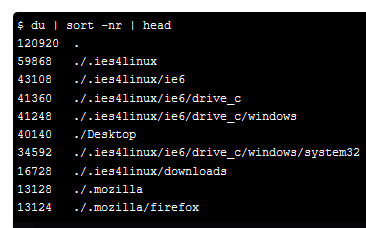
**$**commande << Mots

**Description :** lit depuis le clavier jusqu’à ce que le mots soit écrit.



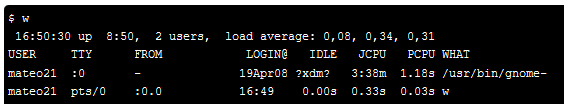
**$**commande **|** commande **|**commande ect..

**Description :** enchaine une suite de commande en renvoyant le résultat à la prochaine commande.



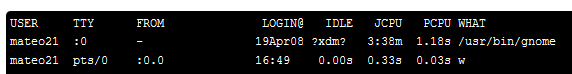
**$**w

**Description :** décris ce que font les utilisateurs connectés.



**$**who

**Description :** décris qui est connecté



**$**ps

**Description :** liste les processus de l’ordinateur en statique.

**Paramètre :**

1. **–ef :** liste tout les processus
2. **–ejH :** Affiche les processus en arbre.
3. **–u :** liste les processus lancé par l’utilisateur.

**$**top

**Description :** liste les processus de l’ordinateur en dynamique.

**$**kill + PID

**Description :** tue un processus via son PID.

**Paramètre :**

1. **–9 :** force l’arrêt.



**$**killall + nom

**Description :** tue un processus via son nom

**$**sudo halt

**Description :** arrête l’ordinateur

**$**sudo reboot

**Description :** redémarre l’ordinateur

**$**&

**Description :** lance un processus en arrière-plan ( petit inconvénient si vous quitter le terminal le processus serra arrêté)



**$**nohup

**Description :** lance un processus en arrière-plan et détaché de la console



**$**CTRL + Z

**Description :** met en pause l’exécution d’un programme

**$**bg

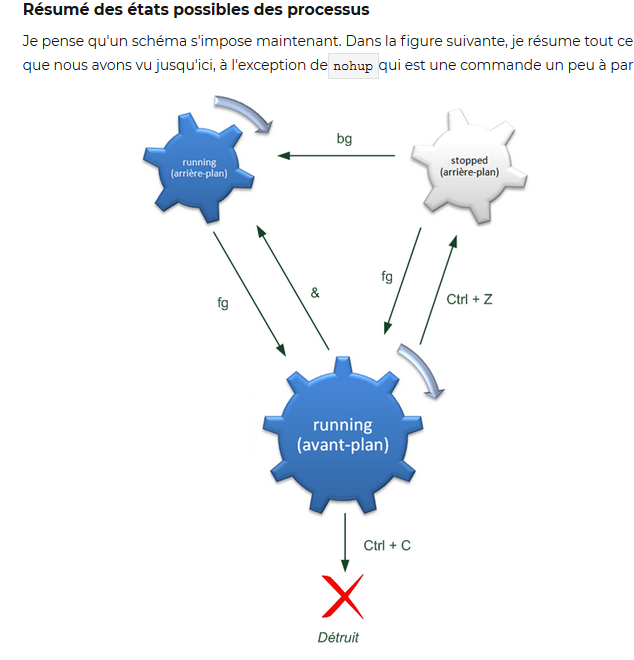
**Description :** passe le processus en arrière plan.

**$**jobs

**Description :** permet de connaître les processus qui tourne en arrière plan.

**$**fg

**Description :** reprend un processus au premier plan.

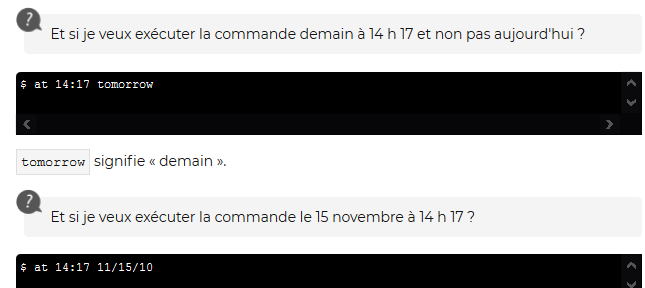


**$**screen

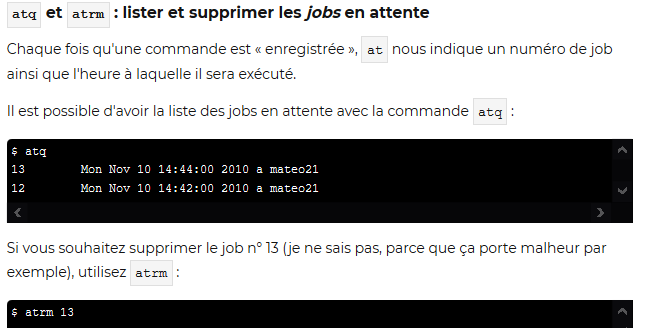
**Description :** permet d’avoir plusieurs console dans une seule console.

**$**at

**Description :** execute une commande plus tard mais une seul fois.







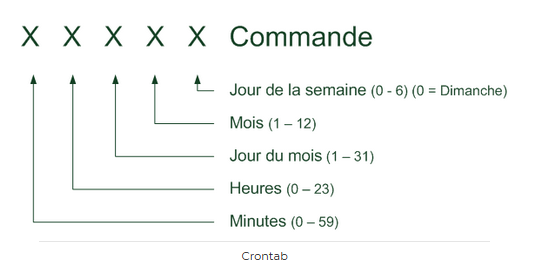


**$**crontab

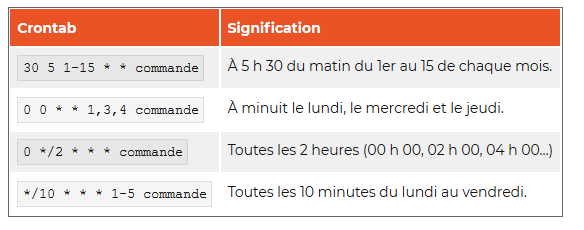
**Description :** execute des commande de manière répétitive.

**Paramètre :**

1. **–l :** Affiche la crontab actuel
2. **–e :** modifie la crontab
3. **–r :** suprimme la crontab







**Commande ; $**sleep + temps ( chiffre + m ou h ou d ) ; commande ;

**Description :** fait une pause entre deux commandes



**$**tar

**Description :** assemble des fichier dans une archive.

**Paramètre :**

1. **–c :** crée une affiche tar



1. **–r :** ajoute un fichier dans l’archive
2. **-x :** extrait les fichiers de l’archive
3. **–v:** affiche les détails des opérations.
4. **–f:** assemble l’archive dans un fichier
5. **–tf:** affiche le contenu sans extraire
6. **–zcvf:** archive et compresse en gzip



1. **–jcvf:** archive et compresse en bzip2
2. **–zxvf:** archive et décompresse en gzip
3. **–jxvf:** archive et décompresse en bzip2



**$**gzip + fichier.tar

**Description :** compresse un fichier



**$**gunzip + fichier.tar.gz

**Description :** décompresse un fichier



**$**bzip2 + fichier.tar

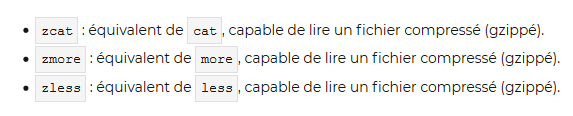
**Description :** compresse un fichier



**$**bunzip2 + fichier.tar.gz

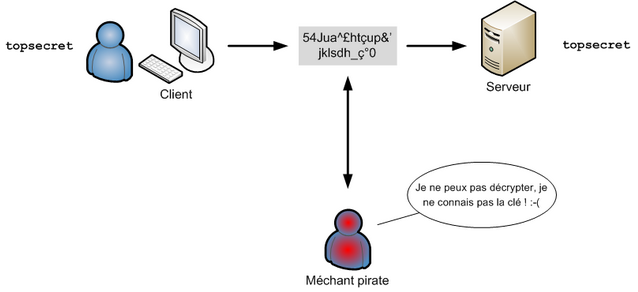
**Description :** décompresse un fichier

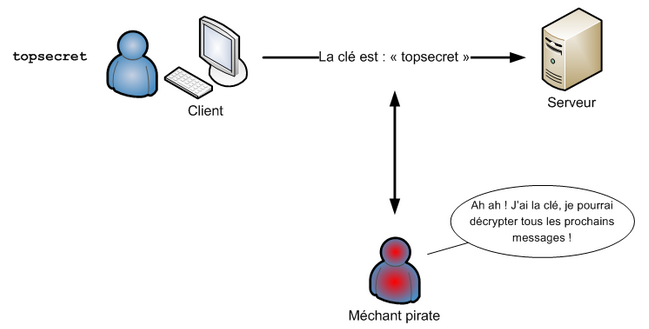




### Connexion via SSH

Le chiffrement symétrique





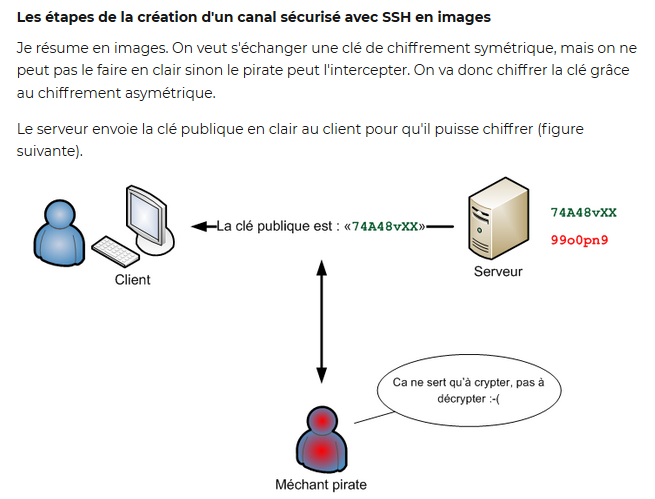
Le chiffrement symétrique est donc puissant, mais il a un gros défaut : il faut communiquer « discrètement » la clé de chiffrement… mais c'est impossible : il faut bien envoyer la clé en clair au début !

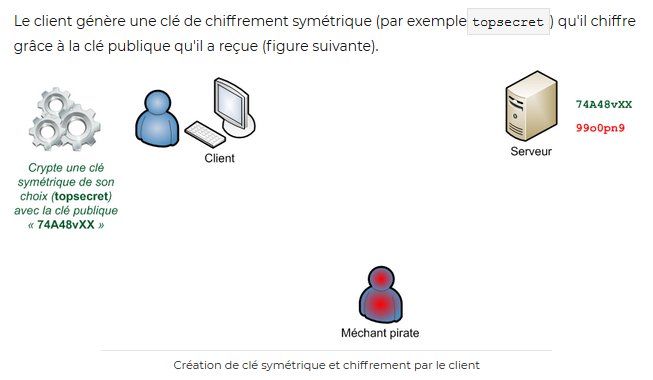
Le chiffrement asymétrique

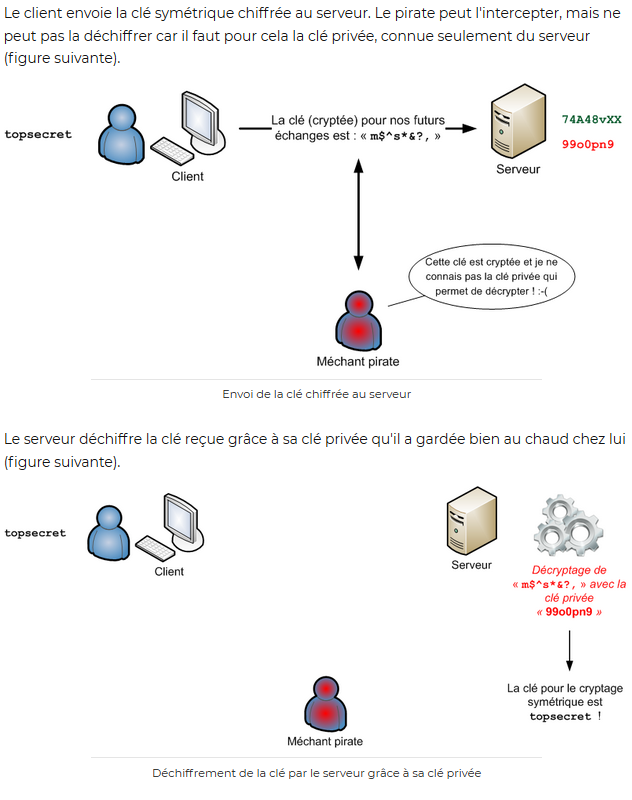
Le chiffrement symétrique utilise une seule clé pour chiffrer et déchiffrer.  
Le chiffrement asymétrique, lui, utilise une clé pour chiffrer, et une autre pour déchiffrer.

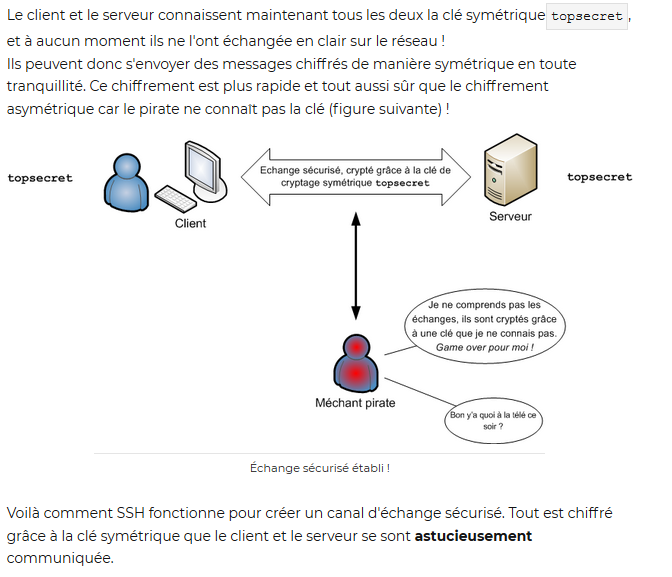
Il y a donc deux clés :

* une clé dite « **publique** » qui sert à **chiffrer**
* une clé dite « **privée** » qui sert à **déchiffrer**.

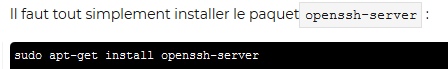


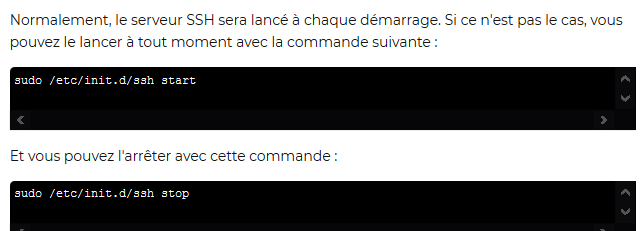






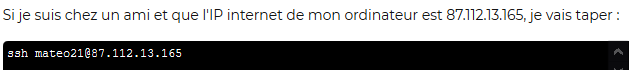
Transformer sa machine en serveur





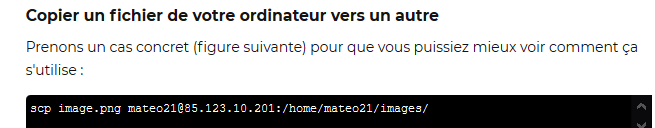
Se connecter via une machine linux

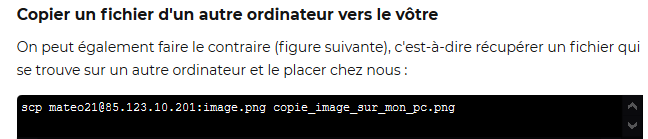


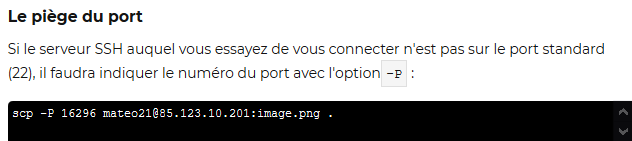


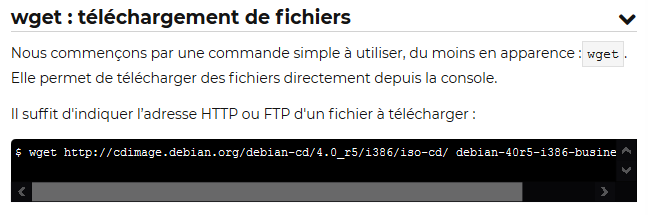


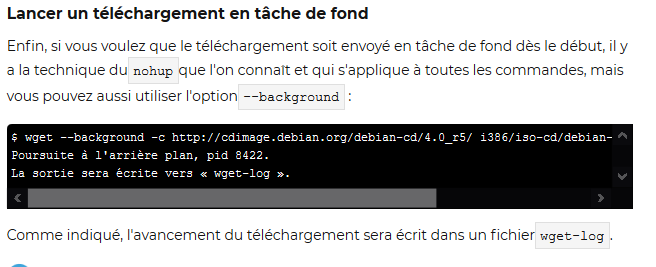


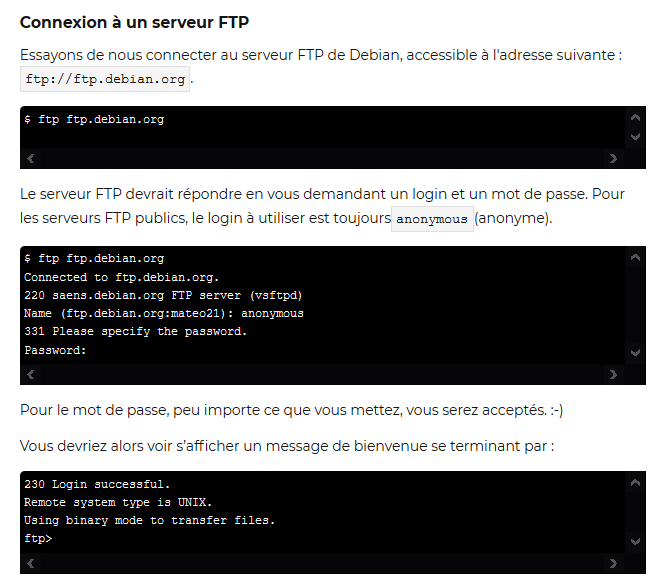










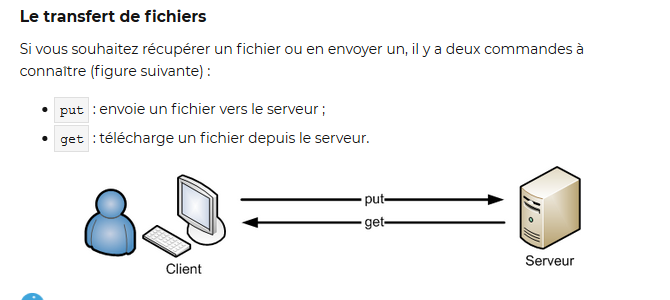


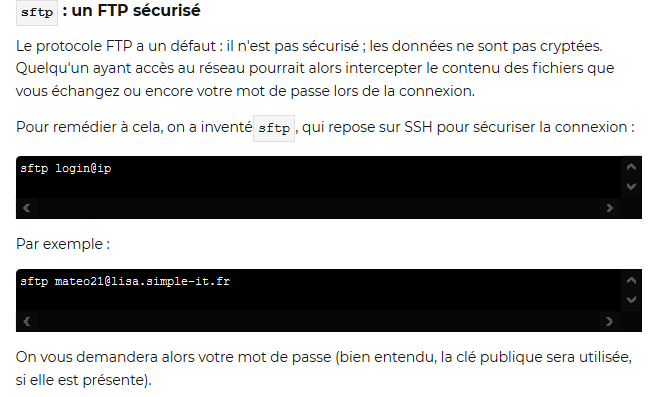
**$**scp + fichier origine + fichier destination

**Description :** Copie un fichier entre deux ordinateurs connectés.









**$ rsync + destinations**

**Description :** Permet la sauvegarde de fichier sur un autre ordinateur comme le cloud ou sur le même ordinateur.

**Paramètre :**

1. **–a :** conserve toutes les informations sur les fichiers, comme les droits (chmod), la date de modification, etc. ;
2. **–r :** sauvegarde aussi tous les sous-dossiers qui se trouvent dans le dossier à sauvegarder ;
3. **–v :** mode verbeux, affiche des informations détaillées sur la copie en cours.
4. **–delete :** permet la suppression des fichiers
5. **–backup :** permet la sauvegarde des fichiers supprimés au cas ou







**$ rsync -arv --delete --backup --backup-dir=/home/mateo21/fichiers\_supprimes Images/ mateo21@IP\_du\_serveur:mes\_backups/**

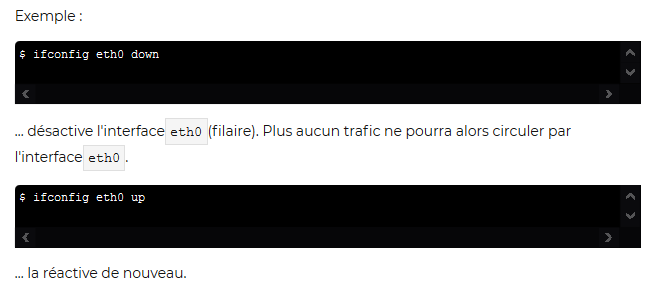
**$ rsync -arv --delete --backup --backup-dir=/home/mateo21/fichiers\_supprimes Images/ mateo21@IP\_du\_serveur:mes\_backups/ -e "ssh -p 12473"**

**$ whois + IP ou host**

**Description :** Permet d’obtenir des informations sur le nom de domaine.

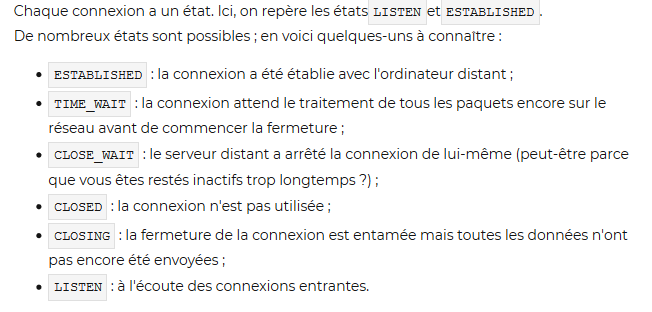






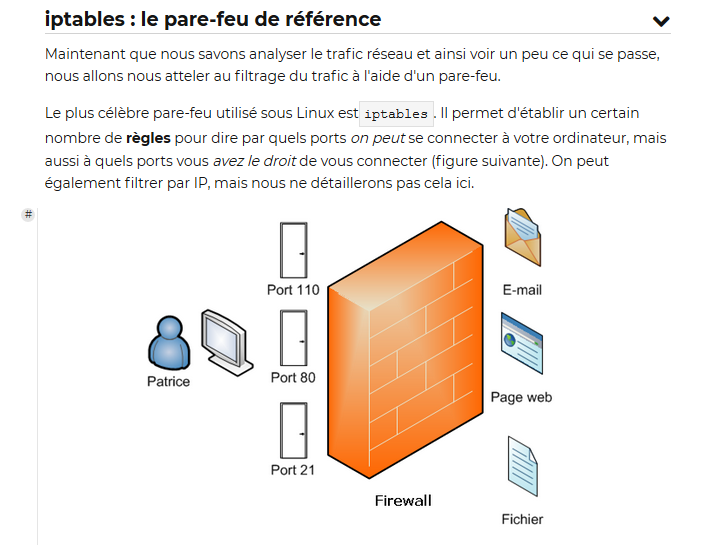




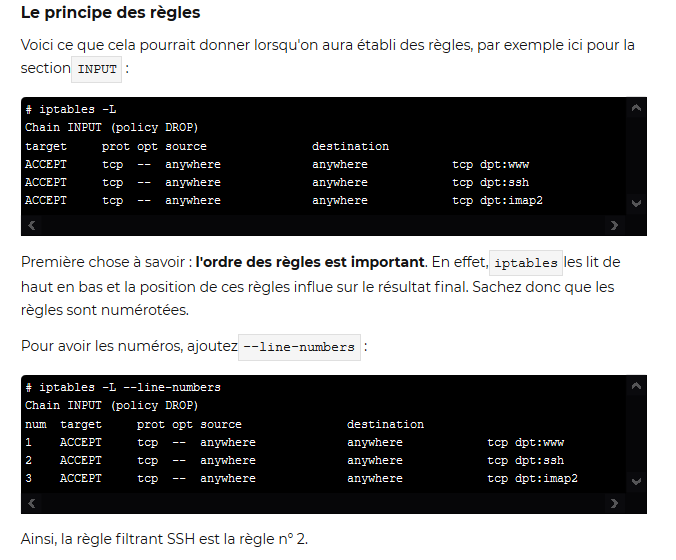


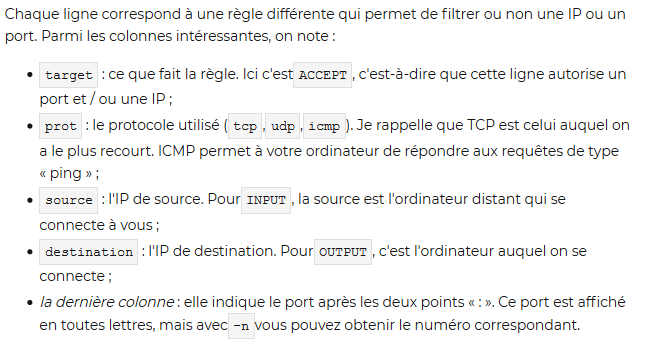


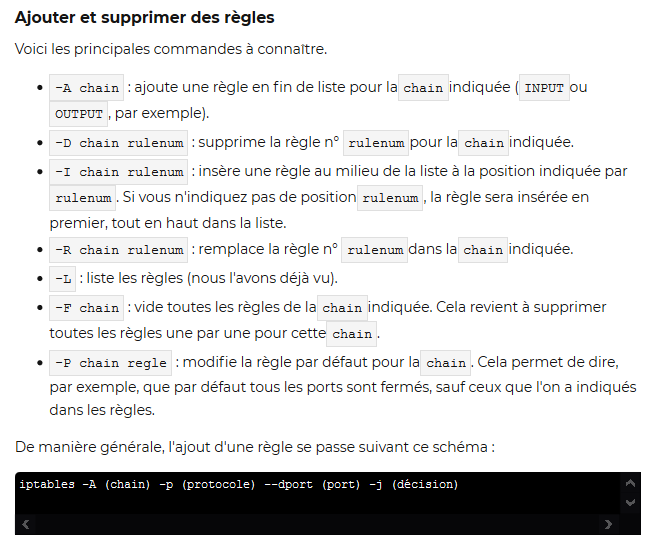






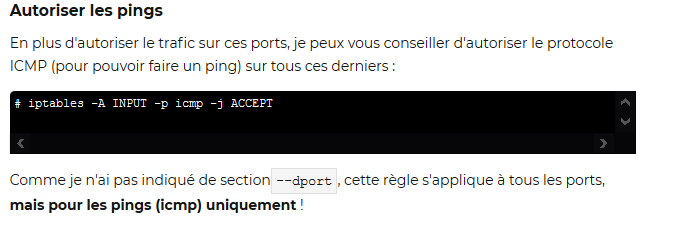
\*

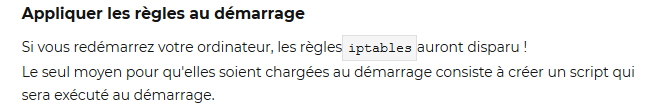


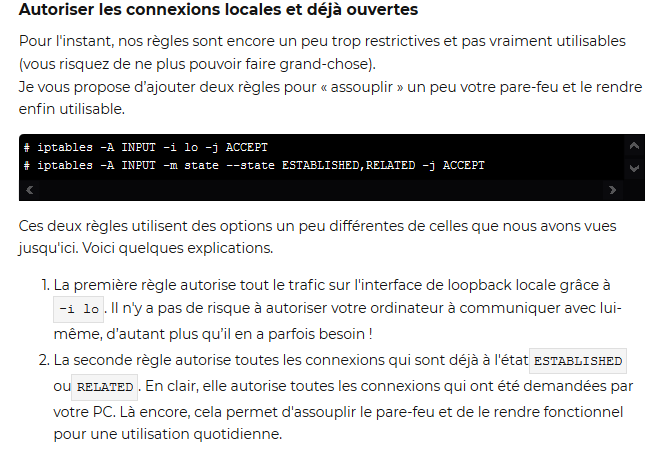


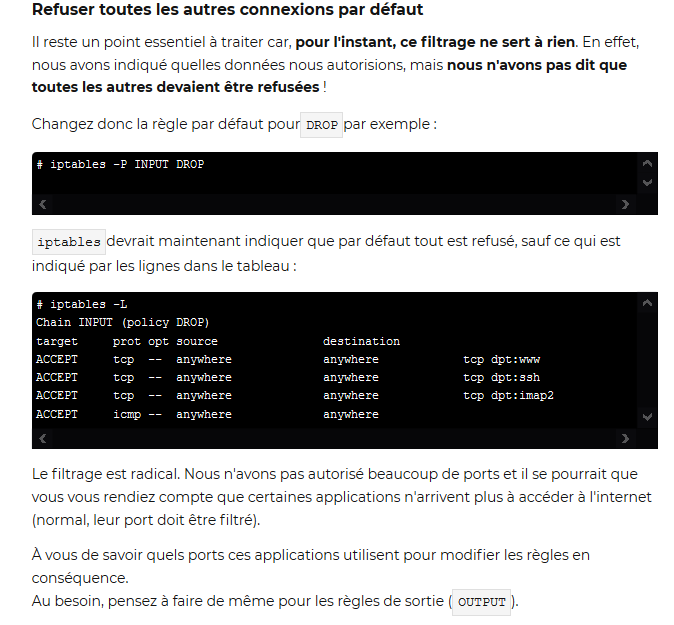










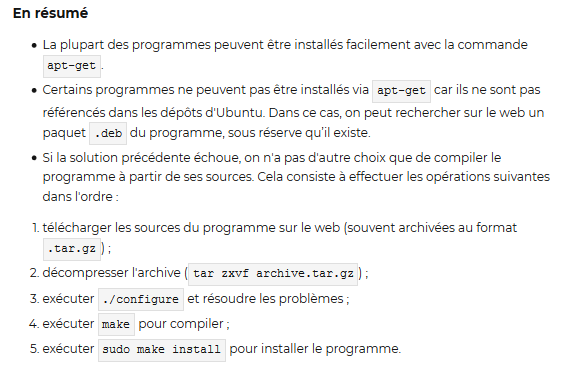


### compiler un programme depuis les sources





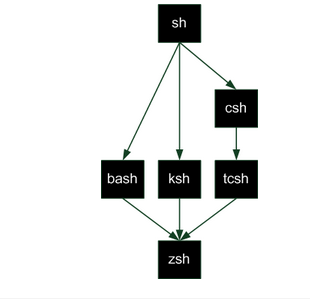


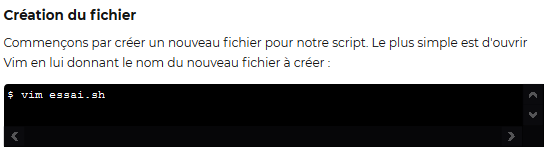


**VIM**

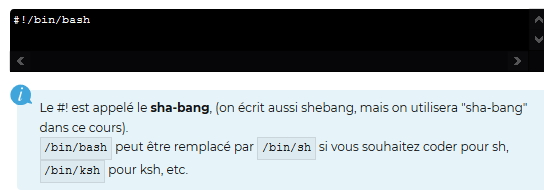
### Les scripts bach

#### Introduction

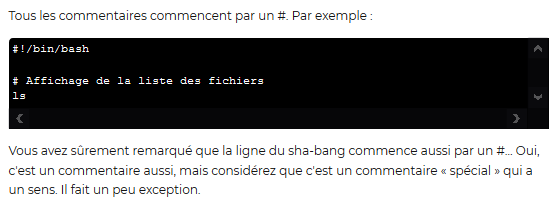






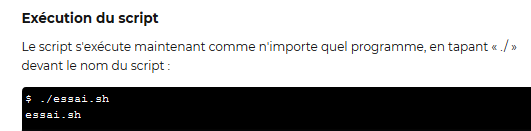


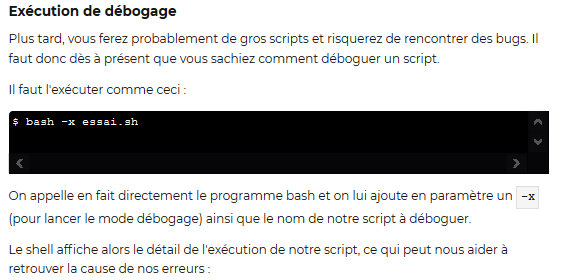


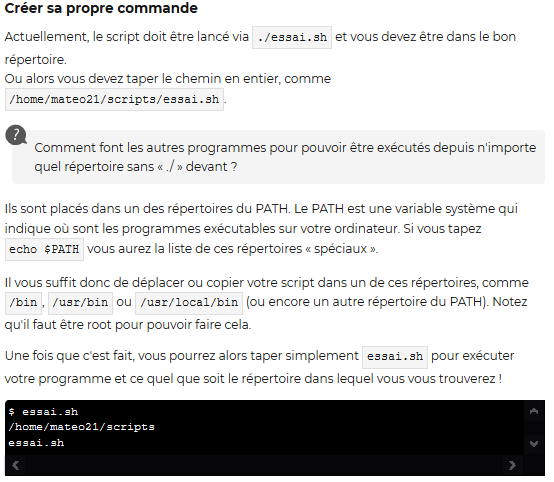






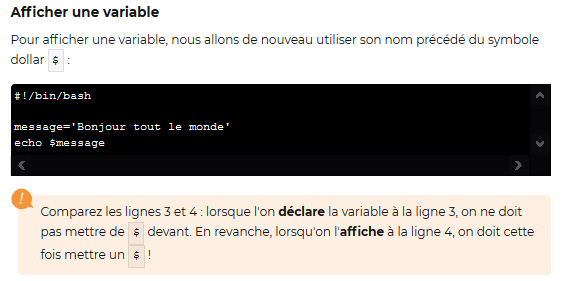


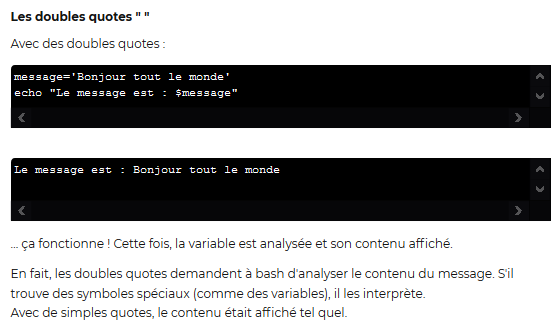


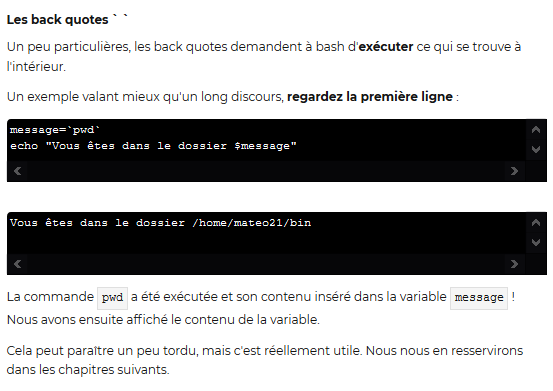


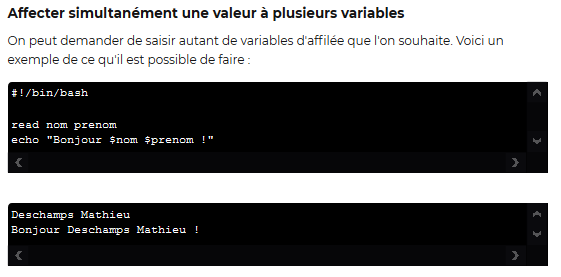
#### Les variables

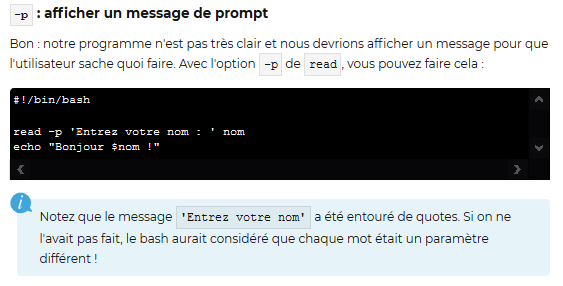


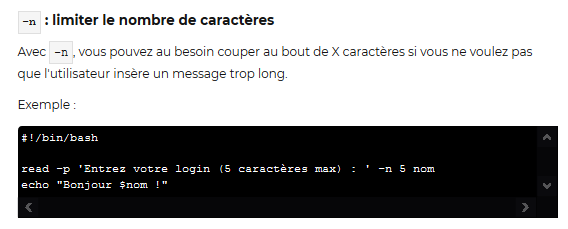






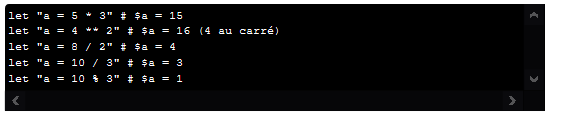




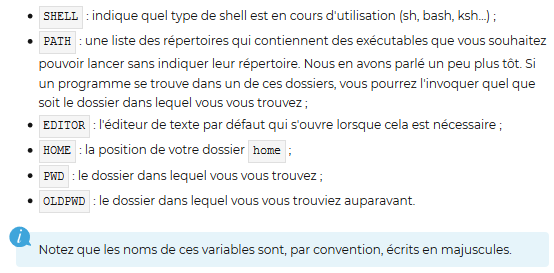


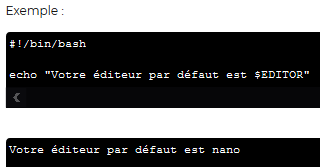


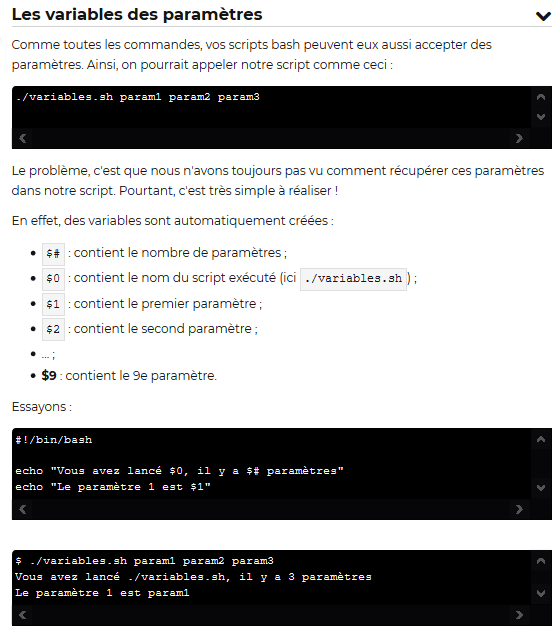


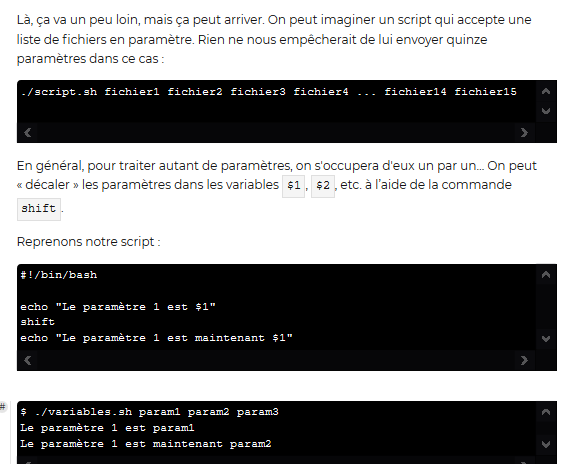


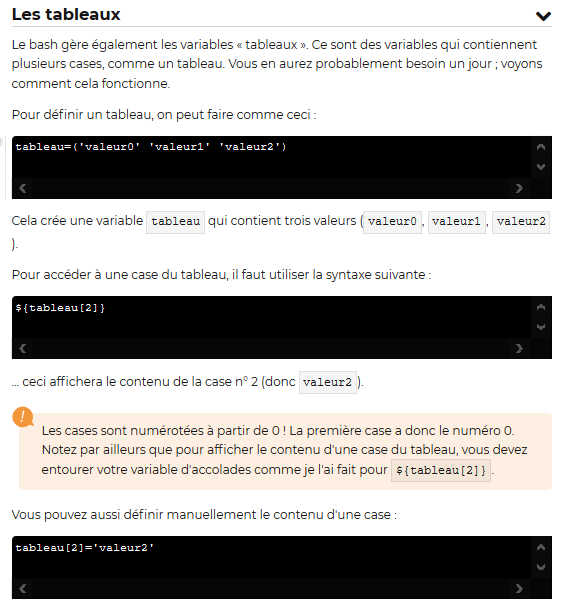


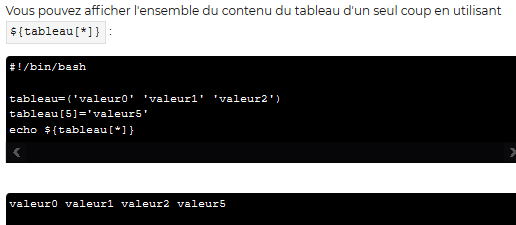




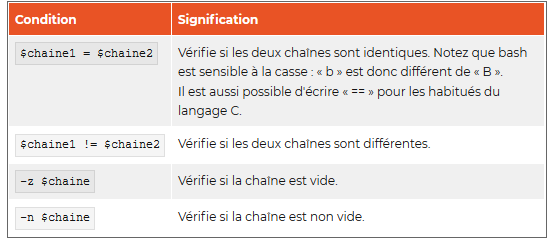


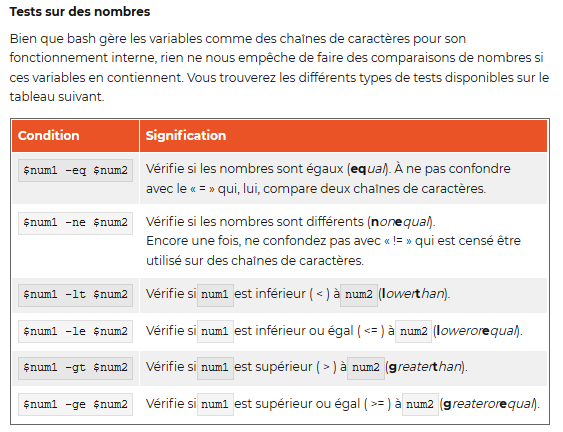




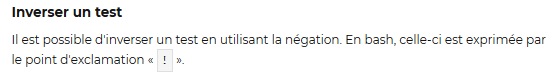
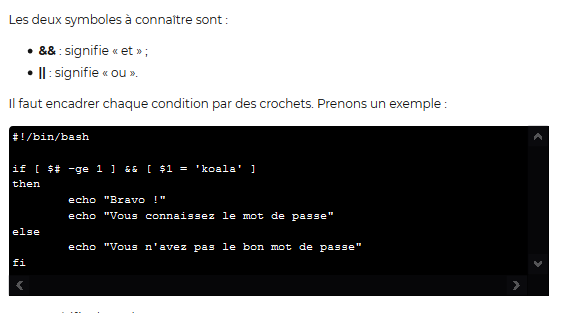


#### Comparaison

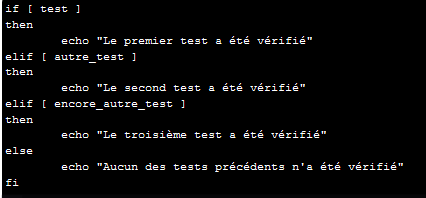


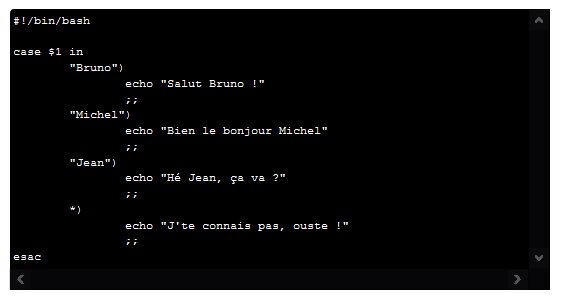


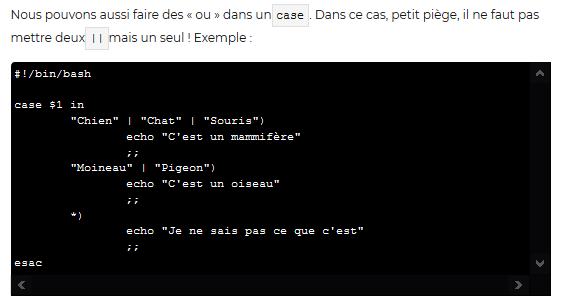




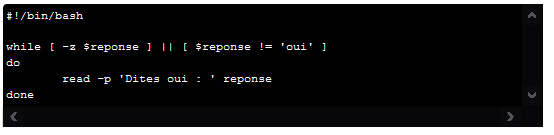
#### Condition IF et case

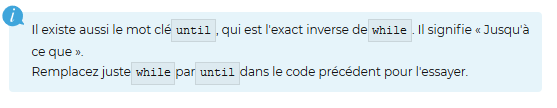






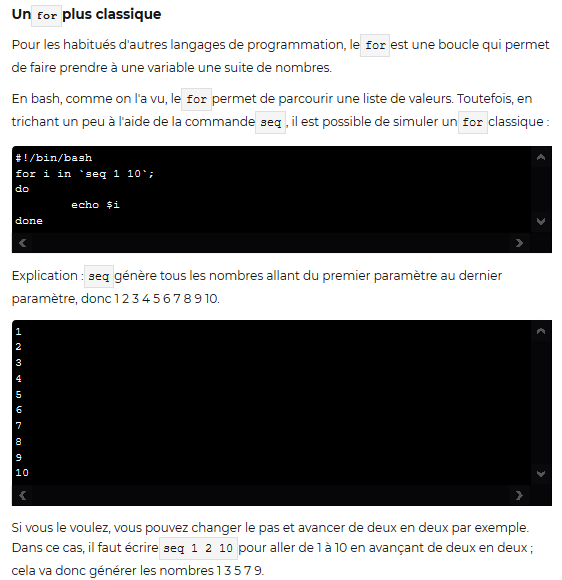
#### Boucle While, until



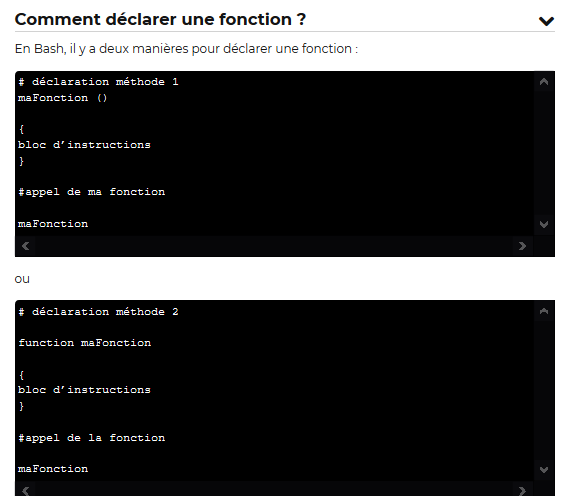


#### Boucle for





#### Les fonctions



### Déclarer une fonction