



Systèmes d'exploitation

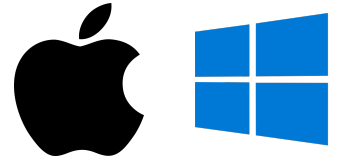


Table des matières

1	Introduction aux systèmes d'exploitation	1
2	Utilisation de LINUX	2
2.1	Utilisation du shell	2
2.2	Quelques particularités de Linux	3
2.3	Application	4

1) Introduction aux systèmes d'exploitation

Voici quelques ressources vidéo :

<https://youtu.be/40hUDAtmAuo> et <https://youtu.be/bdSWj7Y50VY>



FIGURE 1 – Vidéos

Le système UNIX est un système dit "propriétaire" (certaines personnes disent "privateur"), c'est-à-dire un système non libre. Mais plus généralement, qu'est-ce qu'un logiciel libre ?

D'après Wikipédia : "Un logiciel libre est un logiciel dont l'utilisation, l'étude, la modification et la duplication par autrui en vue de sa diffusion sont permises, techniquement et légalement, ceci afin de garantir certaines libertés induites, dont le contrôle du programme par l'utilisateur et la possibilité de partage entre individus". Le système UNIX ne respecte pas ces droits (par exemple le code source d'UNIX n'est pas disponible, l'étude d'UNIX est donc impossible), UNIX est donc un système "propriétaire" (le contraire de "libre"). Attention qui dit logiciel libre ne veut pas forcément dire logiciel gratuit (même si c'est souvent le cas), la confusion entre "libre" et "gratuit" vient de l'anglais puisque "free" veut à la fois dire "libre", mais aussi gratuit.

En 1991, un étudiant finlandais, Linus Torvalds, décide de créer un clone libre d'UNIX en ne partant de rien (on dit "from scratch" en anglais) puisque le code source d'UNIX n'est pas public. Ce clone d'UNIX va

s'appeler Linux (Linus+UNIX). La vidéo ci-dessous raconte l'histoire de Linux en entrant un peu plus dans les détails :

https://youtu.be/IquNF_DXcF8

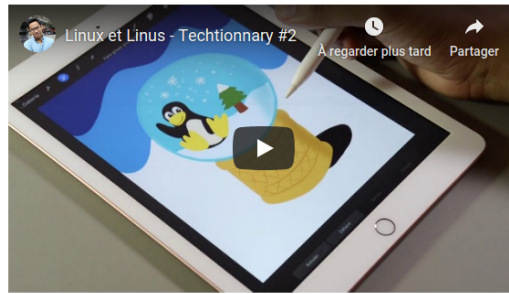


FIGURE 2 – Vidéo

Difficile de parler des systèmes d'exploitation sans parler de Microsoft !

Microsoft a été créée par Bill Gates et Paul Allen en 1975. Microsoft est surtout connue pour son système d'exploitation Windows. Windows est un système d'exploitation "propriétaire", la première version de Windows date 1983, mais à cette date Windows n'est qu'un ajout sur un autre système d'exploitation nommé MS-DOS. Aujourd'hui Windows reste le système d'exploitation le plus utilisé au monde sur les ordinateurs grand public, il faut dire que l'achat de Windows est quasiment imposé lorsque l'on achète un ordinateur dans le commerce, car oui, quand vous achetez un ordinateur neuf, une partie de la somme que vous versez termine dans les poches de Microsoft. Il est possible de se faire rembourser la licence Windows, mais cette opération est relativement complexe.

Enfin pour terminer, quelques mots sur le système d'exploitation des ordinateurs de marque Apple : tous les ordinateurs d'Apple sont livrés avec le système d'exploitation macOS. Ce système macOS est un système d'exploitation UNIX, c'est donc un système d'exploitation propriétaire.

2) Utilisation de LINUX

2.1) Utilisation du shell

Une façon d'interagir avec le système est d'utiliser un terminal (ou console) dans lequel vous pourrez taper des commandes (qui ne sont rien d'autre que des programmes). Le programme avec lequel vous interagissez pour exécuter les commandes s'appelle le shell.

Pour ouvrir une console, il faut aller dans le *menu*, puis *Administration* et *terminal* .

Le shell vous indique qu'il est prêt en affichant une invite (ou prompt), en général le caractère dollar (\$) ou supérieur (>), en début de ligne. Le shell est idéal pour exécuter des commandes « simples » permettant souvent de manipuler des fichiers sur le disque dur et qui affichent leur résultat directement dans le shell (ls, cp, mv,...). Ces commandes, puisqu'elles utilisent le shell pour leurs interactions, ne peuvent être exécutées en dehors d'un terminal.

L'arborescence d'un système GNU/Linux se présente sous la forme suivante :

Remarque : on voit que sous GNU/Linux, un fichier ne possède pas forcément une extension.

Voici une brève description des différents dossiers :

Le système de fichiers racine (root file system), soit le système de fichiers primaire est associé au répertoire le plus haut / :

- /bin commandes binaires utilisateur essentielles (pour tous les utilisateurs)

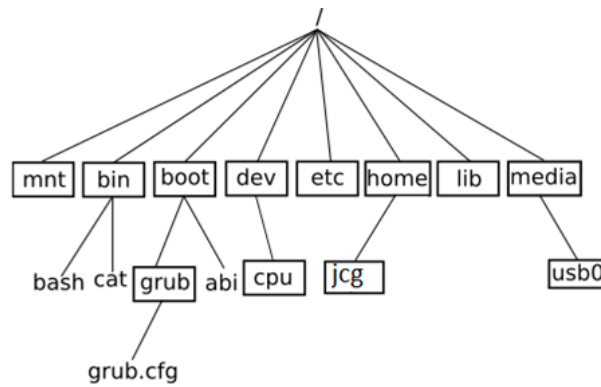


FIGURE 3 – Arborescence

- /boot fichiers statiques du chargeur de lancement /dev fichiers de périphériques
- /etc configuration système spécifique à la machine
- /home répertoires personnels des utilisateurs (optionnel)
- /lib bibliothèques partagées essentielles et modules du noyau
- /mnt point de montage pour les systèmes de fichiers montés temporairement
- /proc système de fichiers virtuel d'information du noyau et des processus /root répertoire personnel de root (optionnel) /sbin binaires système (binaires auparavant mis dans /etc)
- /sys état des périphériques (model device) et sous-systèmes (subsystems)
- /tmp fichiers temporaires

2.2) Quelques particularités de Linux

Avant de commencer, il faut savoir que Linux est sensible à la casse (case sensitive en anglais), c'est à dire qu'il distingue les majuscules des minuscules. Ainsi, pour créer un répertoire, la commande est 'mkdir', ce n'est pas la peine d'essayer MKDIR ou mKdiR, cela ne fonctionnera pas. De même, les noms de fichiers et de répertoires sont également sensibles à la casse.

De plus, sous Unix, les chemins sont séparés par des slash : /etc/init/xfs mais jamais \etc\init\xfs.

Répertoires spéciaux :

- . représente le répertoire courant ;
- .. représente le répertoire parent ;
- ~ (tilde) représente le répertoire maison (home) de l'utilisateur.

Fichiers cachés :

Sous Unix, les fichiers cachés commencent par un point. Par exemple, /.bashrc est un fichier caché, dans le répertoire maison de l'utilisateur, qui contient la configuration de son shell.

Jokers : ? et * :

Les caractères ? et * dans les noms de fichiers et de répertoires permettent de représenter des caractères quelconques. ? représente un seul caractère, tandis que * en représente un nombre quelconque.

Par exemple "*.jpg" représente tous les fichiers se terminant par jpg ; "*toto*" tous les fichiers contenant "toto". Il faut également savoir que c'est le shell qui interprète ces caractères avant de transmettre la ligne de commande. Par exemple, si vous tapez : rm *.tmp, le shell transformera cette ligne de commande en : rm truc1.tmp truc2.tmp truc3.tmp.

Les commandes

Une commande est l'exécution d'un programme dans l'interprète (Shell). Elle prend en entrée des options

et/ou des paramètres. Elle peut renvoyer de l'information à l'écran ou dans un fichier, modifier un fichier, ou produire un message d'erreur.

Une description de toutes les commandes est disponible avec la commande `man` ou `help`. N'hésitez pas à l'utiliser.

Naviguer

- `cd` : se déplacer dans les répertoires.
- `ls` : lister les répertoires et fichiers.
- `pwd` : afficher le répertoire où l'on se trouve exactement.

Déplacer, copier, renommer

- `cp` : copier des fichiers ou des répertoires.
- `mkdir` : créer un dossier.
- `mv` : renommer ou déplacer un fichier ou un dossier.
- `ln` : créer des liens durs et liens symboliques.
- `rm` : supprimer un fichier ou un dossier.
- `rmdir` : supprimer un dossier vide (plus sûr que `rm`).
- `touch` : créer un fichier.

Trouver un fichier

- `find` : recherche pointue !
- `locate` : Recherche accélérée de fichiers ou répertoire.
- `which` : Chemin d'une commande déterminée.

Changer les droits

- `chmod` : gérer la distribution des droits d'un fichier.
- `chown` : gérer la propriété d'un fichier.
- `umask` : création automatique des droits.
- Access Control List : Gestion des droits avancés (experts).

Lire un fichier texte

- `cat` : lire le contenu d'un fichier texte.

Vérifier l'intégrité d'un fichier

- `checksum` : Vérifier l'intégrité d'un fichier.

2.3) Application

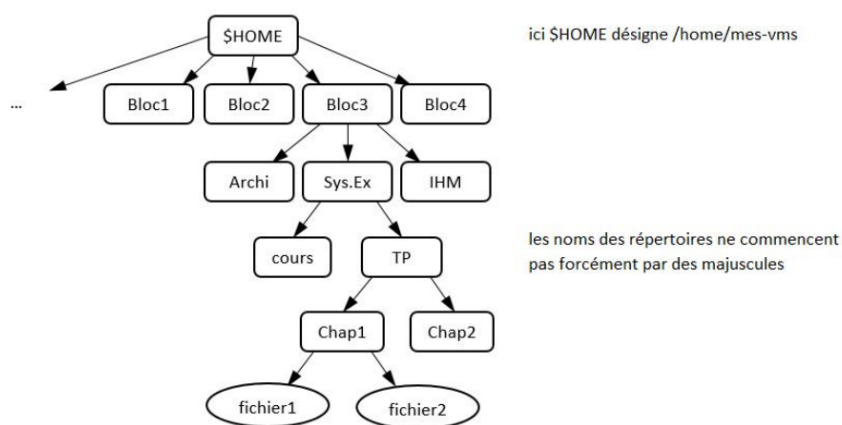


FIGURE 4 – Arborescence

1. A l'aide des commandes `ls`, `cd` et `mkdir`, créer l'arborescence de répertoires indiquée Figure 3.
2. En utilisant la commande `cp`, recopier la structure du répertoire `Sys.Ex` dans les répertoires `Archi` et `IHM`.
3. Déplacer le répertoire `Archi` dans le répertoire `Bloc2`.
4. Copier le contenu du fichier `/etc/group` dans un fichier `toto.txt` dans le répertoire `HOME/Bloc3/Sys.Ex/TP/Chap2/`.
5. Afficher le contenu de ce fichier.
6. Créer un lien depuis le répertoire `HOME` vers ce fichier.
7. Créer un lien symbolique depuis le répertoire `HOME` vers ce fichier. Mettre en évidence ce lien symbolique.
8. En utilisant la commande adéquate, compter le nombre de lignes contenues dans ce fichier.
9. En utilisant le mécanisme de pipelining, compter le nombre de lignes contenant la chaîne `root` dans ce fichier.
10. Créer une archive compressée de tout le contenu de votre répertoire.