

0.1 L'arborescence des fichiers

Un fichier sous Linux (Unix) est une chaîne de caractères non structurée. A tout fichier est associé un bloc d'informations (i-node) décrivant ses propriétés (droits d'accès, propriétaire ...). Ce bloc ne contient pas le nom du fichier. Chaque fichier est identifié par un index (nombre entier) unique pour chaque fichier. la désignation d'un fichier se fait par l'intermédiaire de fichiers particuliers : les répertoires. Un répertoire est un fichier formé d'un ensemble de couples (index, nom de fichier) permettant l'accès aux fichiers. Un premier répertoire sans nom et d'index 2 constitue la racine du système de fichiers. Il va contenir des fichiers non-répertoires et/ou des répertoires qui à leur tour vont contenir des fichiers non-répertoires et bien-sûr d'autres répertoires qui à leurs ... L'ensemble forme une arborescence dont les feuilles sont des fichiers non-répertoires ou des répertoires vides. Linux traite par ailleurs tous les périphériques (clavier, souris, disques, etc ...) comme des fichiers spéciaux intégrés dans l'arborescence (dans le répertoire /dev).

0.1.1 1. L'arborescence :

Ainsi votre disque dur /dev/hda est un fichier. Chaque répertoire sous Unix à une fonction particulière, voyons les plus importants :

- **/** : C'est la racine du système, un raccourci facile et erroné serait de dire qu'il s'agit du C : de windows ® . C'est faux, car sous Windows C : \Winnt est un répertoire contenu dans la partition C : \ hors sous Linux le répertoire /boot bien qu'il semble rattaché à /, n'est pas forcément sur la même partition que ce dernier. / est le répertoire à partir duquel tous les répertoires sont repérés, qu'ils soient distants, locaux ou situés sur une autre partition.
- **/bin** : Contient les commandes de bases utilisables à partir du shell (grep, mount, cp ...).
- **/boot** : contient les informations nécessaires au démarrage de la machine.
- **/dev** : contient les fichiers spéciaux correspondant aux périphériques.
- **/etc** : la plupart des fichiers de configuration.
- **/home** : Les C : \Documents and settings\ de windows NT ®(XP/2000) et C : \Mes documents\ de windows ® 9x en sont des versions édulcorées. Il contient les répertoires personnels des utilisateurs, par exemple /home/jack est, le répertoire personnel et privé de l'utilisateur jack. C'est le seul endroit qui lui appartienne.
- **/lib** : contient les principales bibliothèques partagées (équivalent aux DLL de Windows ®).
- **/lost+found** : c'est là que sont entreposés les résultats des scandisk quand ils ont lieu.
- **/mnt** : les répertoires utilisés pour monter temporairement un système de fichiers. /mnt/floppy pour la disquette, /mnt/cdrom pour le cdrom ou /mnt/windows_c pour la partition C : de windows.
- **/opt** : Permet de reproduire une arborescence /usr complète à partir d'un nouveau répertoire. Par exemple si j'installe kde à partir des sources dans /opt/kde. Il va se créer des répertoires /opt/kde/bin/, /opt/kde/share ... Plus tard pour désinstaller kde je n'aurais qu'à supprimer /opt/kde.
- **/proc** : un répertoire factice, dont les fichiers contiennent des infos sur l'état du sys-

tème et des processus en cours d'exécution.

- **/root** : le répertoire personnel et privé de l'administrateur système : root.
- **/sbin** : Contient les commandes de base nécessaires à l'administration système (fsck, mke2fs ...)
- **/tmp** : C'est l'équivalent des C :\WINNT\Temp\ de windows ® NT/2000/XP et C :\WINDOWS\Temp de windows ® 9x
- **/usr** : C'est l'équivalent de C :\Programme Files\ Mais en plus ordonné.
- **/usr/X11R6** : Tout ce qui concerne X-Window. Organisation similaire à /usr.
- **/usr/bin** : les exécutables (ils n'ont pas l'extension .exe)
- **/usr/doc** : la documentation sur les applications installées.
- **/usr/etc** : des fichiers de configuration, qui ne sont pas dans /etc/
- **/usr/games** : Le répertoire d'installation de jeux.
- **/usr/include** : les fichiers d'en-tête pour la programmation.
- **/usr/info** : la doc au format GNU info.
- **/usr/lib** : les bibliothèques partagées (DLL) non vitales.
- **/usr/local** : une sous-hiérarchie qui contient des logiciels compilés sur place à partir des sources. Organisation similaire à /usr.
- **/usr/man** : le manuel en ligne. Les fichiers sont compressés.
- **/usr/sbin** : Des exécutables pour l'administration réseau et système.
- **/usr/share** : Des fichiers de données.
- **/usr/spool** : Un lien symbolique vers /var/spool.
- **/usr/src** : les sources de certains logiciels, principalement le noyau de Linux (/usr/src/linux)
- **/var** : des données fréquemment réécrites, comme les logs.
- **/var/catman** : les pages du manuel décompressées (ça se fait à la demande).
- **/var/lib** : Des bases de données, des fichiers de configuration...
- **/var/local** : complète /usr/local de la même façon que /var complète /usr.
- **/var/lock** : Des fichiers qui servent à marquer l'utilisation de certaines ressources. Par exemple, quand un logiciel se sert du modem, il crée un fichier ici pour le signaler.
- **/var/log** : le journal (log) du système. Notamment /var/log/messages par lequel le noyau vous tient son monologue.
- **/var/run** : principalement des infos sur les serveurs en fonctionnement.
- **/var/spool** : les spools : tout ce qui est "de passage" en attendant d'être utilisé par un logiciel. Ca inclut entre autres le mail, les news, les files d'attente des imprimantes...
- **/var/tmp** : La corbeille de /var.

0.1.2 2. Les liens :

Un lien vers un fichier est simplement un pointeur qui permet d'accéder à un fichier ou répertoire en utilisant un chemin ou nom différent. La commande qui permet de créer des liens est **"ln"**.

0.1.3 2.1 Les liens physiques :

Sous linux, un même fichier peut avoir plusieurs noms, c'est ce qu'on appelle un "lien physique" ou "lien dur". On les crée avec la commande :

In original lien

À partir de ce moment, original et lien désigne le même fichier (il n'y a pas duplication), une modification de l'un entraîne la modification de l'autre.

0.1.4 2.2 Les liens symboliques :

Malheureusement les liens physiques ont beaucoup de limitations et posent des problèmes de bouclage, lorsqu'ils sont utilisés sur des répertoires, c'est pourquoi je vous conseille de toujours utiliser des liens symboliques : On crée un lien symbolique par :

In -s original lien

Le fichier lien est un alias, sa suppression n'interfère pas sur original, mais les modifications apportées à lien sont automatiquement répercutées sur original. Si vous supprimez original, lien devient un "**lien mort**".