

Le système de fichiers UNIX

Etienne Lozes

ENS Cachan

Différents types de fichiers

Fichier = objet typé admettant lecture et écriture.

Quelques types de fichiers

- fichier régulier (fichier disque "classique")
- répertoire
- fichiers spéciaux (périphériques, disques, partitions,...)
mode bloc (cache) ou caractère
- tubes
- sockets (en particulier réseau)
- liens symboliques

Du point de vue utilisateur/programmeur : pas de différence d'utilisation.

Les i-noeuds (noeuds d'index)

i-noeud = référence interne d'un fichier.

Système de fichiers = table de i-noeuds (+ arborescence)

Informations contenues dans un i-noeud

- type
- droits d'accès et propriétaire (utilisateur, groupe)
- taille
- nombre de liens (références externes)
- des dates
- adresse des blocs utilisés (fichiers disques)
- ressource associée (fichiers spéciaux)

⇒ Démo

Remarque : pas de nom de fichier dans la table des i-noeuds

L'arborescence

Désignation externe : des i-noeuds par un *chemin d'accès* dans un arbre.

Caractéristiques

Racine absolue : `/`.

Désignation relative (au répertoire de travail du programme) :
`rep1/rep2/.../nomdufichier` ou absolue (en commençant par `/`)

Répertoires `.` et `..`

Questions

- 1 Est-il possible d'avoir plusieurs chemins d'accès pour un même fichier (i.e. i-noeud) ?
- 2 Donner au moins trois façons de désigner le fichier `monprog` depuis le répertoire de travail `/home/lozes`. Comment exécuter ce programme ?

Quelques répertoires particuliers

Remarque : pas standard, varie suivant les systèmes.

- /bin, /usr/bin : exécutables standards (ls, pwd,...)
- /lib, /usr/lib : librairies standards (stdlib, stdio,...)
- /dev : arborescence des fichiers spéciaux de périphériques
- /tmp : fichiers temporaires
- /etc : fichiers infos système (ex : mots de passe, table de montage, impression...)
- /mnt : répertoire des points de montage

Montage : greffe d'un système de fichier à un niveau de l'arborescence (ex : NFS, stockage amovible,...)

Mode d'un fichier, contenu des 12 bits

Le droit d'accès à un fichier dépend de son propriétaire, de son groupe et du mode d'accès :

Les 9 bits "ugo"

Pour chaque identité :	droit d'accès ou non en
• propriétaire (u)	• lecture (r)
• groupe (g)	• écriture (w)
• autre (o)	• exécution (x)

Questions

- 1 Quels droits faut-il pour modifier les droits ?
- 2 Comment cacher des fichiers aux autres en les laissant consultables ?
- 3 Peut-on effacer un fichier sans avoir son droit en écriture ?

Mode d'un fichier, contenu des 12 bits

Les 3 bits "exotiques"

- le set-uid bit et le set-gid bit : change l'id de l'utilisateur à l'exécution
- le sticky bit : à l'origine = maintien en Swap, POSIX 2001 = protection d'effacement dans un répertoire

Questions

- 1 Donner un exemple de fichier ayant le set-uid bit activé.
- 2 Qu'est-ce qui peut provoquer l'effacement (définitif) d'un fichier ?