

TP n°2: Gestion des fichiers

Objectifs

A l'issue de ce chapitre, vous connaîtrez la structure des systèmes de fichiers, leur gestion.

Contenu

L'arborescence des fichiers
Les principaux répertoires
Les types de fichiers, les droits
La gestion de l'arborescence
Les attributs de fichiers

1- Rappels de cours: Structure des fichiers /etc/passwd et /etc/group

1.1- Les principaux répertoires

- Les principaux répertoires:

- / : Répertoire racine, là où tous les autres répertoires sont montés (accrochés)
- /bin : commandes UNIX, une partie des binaires du système et quelques commandes
exemple : ls, date, who
- /sbin : programmes exécutables indispensables à la gestion du système.
- /etc : quelques fichiers de configuration et des fichiers systèmes pour le démarrage
exemple : /etc/shutdown, /etc/init, /etc/passwd, /etc/group
- /dev : fichiers unité (périphériques, spéciaux)
exemple : /dev/lp0 imprimante 0
- /home : partie où sont stockés les fichiers propres aux utilisateurs
- /var : fichiers temporaires de taille variable de quelques démons, de spools d'email et d'imprimantes, de logs, de locks ...
- /opt : lieu d'installation préféré des logiciels "modernes"
- /boot : image du noyau pour Linux
- /tmp : (temporary) fichiers temporaires, utilisés par l'éditeur de texte vi, les compilateurs...
- /usr : espace "standard"
- /usr/bin : pour les binaires
- /usr/lib : (library) fichiers d'information, pour les bibliothèques du langage C
- /usr/include : fichiers d'entête pour programmes C (.h)
- /usr/local : espace "non std", personnalisation locale du système
- /usr/local/bin : rajout de binaires en local
- /usr/local/lib : idem pour les bibliothèques
- /usr/local/include : idem pour les "includes"
- /usr/local/src : code source des différents programmes du système
- /usr/man : aide en ligne
- /mnt : (mount) montage de disquettes, donc la possibilité d'accéder aux données présentes dans la disquette à partir du répertoire /mnt, en utilisant les commandes d'UNIX
- /lost+found : (perdu et trouvé) contient les fichiers retrouvés par la commande fsck
fsck : vérifie l'intégrité des données dans un SdF

- Type de fichier:

La commande « file »

- Visualiser un fichier:
Les commandes « cat », « more » ou « less »
- Dump d'un fichier:
La commande « hexdump »
La commande « od »

2- Activités de préparation au TP

2.1- Les types de fichiers

- Classification

Fichier	Symbole (ls -l)	Création	Destruction
Ordinaire	-	vi, ...	rm
Répertoire	d	mkdir	rmdir, rm -r
Périphérique caractère	c	mknod	rm
Périphérique bloc	b	mknod	rm
Socket locale	s	socket(2)	rm
Tube nommé	p	mknod	rm
Lien symbolique	l	ln -s	rm

- Unités de disque:

Emplacement des unités: /dev

Fichiers son: /dev/audio (c), /dev/sbpcd (b)

Unité de CD-ROM: /dev/hdc (b)

Console: /dev/console (c)

Ports de modems: /dev/cua0 (c)

Unités de disquette: /dev/fd0 (b)

Unités à bandes: /dev/rft0, /dev/nrtf0 (b)

- Exemples:

```
# file /bin/bash
```

```
/bin/bash: ELF 32-bit LSB executable, Intel 80386, version 1, dynamically linked,  
stripped
```

```
# file /etc/passwd
```

```
/etc/passwd: ASCII text
```

```
# tail -2 /etc/passwd > fichier
```

```
# cat fichier
```

```
ali:!!:500:500::/home/ali:/bin/bash
```

```
brahim:tbiYDmgxAcKi2: 65536:65536::/home/brahim:/bin/bash
```

hexdump -c /etc/passwd # -c : mode caractère

```
00000000  a l i l 2 3 : ! ! : 5 0 0 : 5 0
0000010  0 : : / h o m e / a l i l 2 3 :
0000020  / b i n / b a s h \n a l i l : t
0000030  b i Y D m g x A c K i 2 : 6 5 5
0000040  3 6 : 6 5 5 3 6 : : / h o m e /
0000050  a l i l : / b i n / b a s h \n
```

od -xc fichier # -x : hexadécimal -c : mode caractère

```
00000000  6970 7265 6572 213a 3a21 3035 3a30 3035
        a l i l 2 3 : ! ! : 5 0 0 : 5 0
0000020  3a30 2f3a 6f68 656d 702f 6569 7272 3a65
        0 : / h o m e / a l i l 2 3
0000040  622f 6e69 622f 7361 0a68 6170 6c75 743a
        / b i n / b a s h \n a l i l : t
0000060  6962 4459 676d 4178 4b63 3269 363a 3535
        b i Y D m g x A c K i 2 : 6 5 5
0000100  3633 363a 3535 3633 3a3a 682f 6d6f 2f65
        3 6 : 6 5 5 3 6 : : / h o m e /
0000120  6170 6c75 2f3a 6962 2f6e 6162 6873 630a
        a l i l : / b i n / b a s h \n
```

2.2- Les droits

- Les neuf droits fondamentaux (valeurs octales : 400,200,100,40,20,10,4,2,1)

- Fichier ordinaire

- « read » : lire les octets du fichier (autorise par exemple la copie du fichier.)
- « write » : ajouter, retirer ou modifier des octets.
- « execute » : considérer le fichier comme une commande

- Répertoire

- « read » : connaître la liste des fichiers du répertoire (exécution, avec au plus l'option « -i », de la commande ls).
- « write » : créer ou de supprimer des fichiers d'un répertoire (nécessite obligatoirement le droit « x »).
- « execute » : accéder aux fichiers d'un répertoire. (clé indispensable pour que les droits d'accès d'un fichier soient contrôlés). A défaut, aucune opération n'est possible sur le fichier, quels que soient les droits de l'utilisateur. Le droit d'exécution est aussi nécessaire pour qu'un répertoire devienne le répertoire courant, grâce à la commande cd.

- Le « sticky bit » (valeur octale : 1000, valeur symbolique : lettre t)

- Exécutable: il reste en mémoire, son chargement est rapide
- Répertoire: la destruction d'un fichier est réservée au propriétaire

- Les droits d'endossement (valeurs octales : SUID=4000, SGID=2000, valeur symbolique : s)

- Exécutable

- SUID : le processus possède les droits du propriétaire du programme
- SGID : le processus possède les droits du groupe du programme
- exemple:

```
# ls -l /usr/bin/passwd
```

```
-r-sr-xr-x 1 root root 12345 oct 2 2001 /usr/bin/passwd
```

```
ls -l /usr/bin/lpr
```

```
-r-sr-sr-x 1 root root 15068 oct 2 1998 /usr/bin/lpr
```

```
# ls -ld /var/spool/lpd/epson640
drwxr-xr-x 2 root lp 1024 avr 30 11:08 /var/spool/lpd/epson640
```

```
$ lpr fichier # commande exécutée par un utilisateur ali,
# d'UID réel 'ali' et de GID réel 'stage',
# mais d'UID effectif root et de GID effectif
```

- Répertoire (SGID)

Les fichiers créés dans le répertoire appartiennent au groupe du répertoire et non au groupe de l'utilisateur qui les crée

```
$ ls -ld repertoire
drwxrwsr-x 2 ali famille 1024 oct 10 10:15 repertoire
$ id
uid=500(ali) gid=500(ali) groups=500(ali)
$ cat > repertoire/f
Quel est le groupe
^D
$ ls -l repertoire/f
-rw-rw-r-- 1 ali famille 19 Oct 10 10:18 repertoire/f
```

2.3- Les commandes de gestion de l'arborescence

- Principales commandes

ls -lR: Affiche arborescence & caractéristiques
du: Affiche une arborescence et/ou sa taille
rm -Rf: Détruit une arborescence
cp -Rfp: Copie une arborescence
chmod -R: Change les droits des fichiers d'une arb.
chgrp -R: Change le groupe des fichiers d'une arb.
chown -R: Change le propriétaire/groupe des fichiers
find : Effectue une recherche sur une arborescence

- La commande find

Structure et technique de travail:

3 indications

A partir de quel répertoire commencer la recherche
Quels sont les critères de recherche à mettre en œuvre
Que doit-il se passer si un fichier répond à ce critère

Syntaxe:

find repertoire [-critère [argument_critère]] ...

Critères de sélection:

- name nom du fichier
- type f, d, c, b, p, s, l
- size +-valeurcbk : taille >, < à valeur (car, blocs, ko)
sans signe, par défaut = taille
- user propriétaire
- group groupe
- perm +-droits : au plus/moins les droits (rwx)
- ctime nbjours : status fichier modifié depuis nbjours

-mtime nbjours : dernière modif. remonte à nbjrs
 -atime nbjours : dernier accès remonte à nbjrs
 -links

Critères d'exécution:

-print	Affiche le chemin d'accès
-exec cmde {} \;	Exécute cmde avec comme argument le fichier
-ok cmde {} \;	Demande une confirmation pour exécuter la cmde avec comme argument le fichier

• Exemples:

```
# du-h
4.0k  ./soft-jf/CVS
9.0k  ./soft-jf
10k
```

```
# df-h
File system  Size  Used Avail Use% Mounted on
/dev/hda6    288M 134M 139M 49% /
/dev/hda2    46M  2.6M 41M  6%  /boot
...
usbdevfs     737M 737M  0    0%  /mnt/f
```

```
# ls -l fic*
-rw-r--r-- 1 ali ali 146 avr 30 12:29 fichier
-rw-r--r-- 1 ali ali 1044 avr 30 11:06 fic.doc
```

```
# chown brahim fichier      # ne change que le propriétaire
# ls -l fichier
-rw-r--r-- 1 ali brahim 146 avr 30 12:29 fichier
```

```
# chown ali.projet fic.doc   # change le propriétaire et le groupe
# ls -l fic.doc
-rw-r--r-- 1 ali projet 1044 avr 30 11:06 fic.doc
```

```
$ find . -name archive -print # rechercher tous les fichiers archive
./outils. copie/outils/archive
./outils/archive
$ find /home -name '*.c' -print # afficher les fichiers dont le suffixe est .c
$ find . -type d -print        # afficher les noms des répertoires seulement
./outils
./outils/archive
```

```
$ ls -l f*          # le groupe de f3 n'est pas users
-rw-r--r-- 1 ali users 24 Jun 11 16:51 f1
-rw-r--r-- 1 ali users 21 Jun 11 16:51 f2
-rw-r--r-- 1 ali adm 271 Oct 23 1994 f3
$                  # rechercher les fichiers dont le groupe n'est pas users
$ find /home/ali ! -group users -print
/home/ali/f3
$                  # rechercher les fichiers modifiés aujourd'hui
$ find /home -mtime 0 -print
```

```
$                  # rechercher les fichiers qui n'appartiennent pas à ali,
```

```
# dont la taille est supérieure
# à 1000 octets ou le dernier accès remonte à moins de 30 jours
$ find /home ! -user ali \( -size +10000c -o -atime -30 \)

$ # rechercher et détruire tous les fichiers réguliers dont la taille est nulle
$ find /home -type f -size 0 -exec rm -f {} \;
```

2.4- Les attributs des fichiers

• Les principaux attributs

```
a      :   Fichier log
i      :   Fichier non modifiable
s      :   Fichier physiquement détruit
S      :   Fichier synchrone
```

Les attributs d'un fichier sont des caractéristiques supplémentaires. qui viennent s'ajouter, dans le système de fichiers «ext2» aux caractéristiques habituelles.

Les attributs:	Description:
A	L'heure et la date de dernier accès (« access time ») ne sont plus modifiées si cet attribut est positionné, ceci par souci de performance.
A	Un fichier qui possède cet attribut ne peut, en écriture, qu'être ouvert en ajout.
C	Les écritures dans le fichier sont automatiquement compressées et les lectures décompressées.
D	Le fichier ne sera pas sauvegardé par la commande dump.
i	Le fichier ne peut pas être modifié, détruit, renommé et il est impossible de créer des liens sur ce fichier. Seul l'administrateur root peut positionner cet attribut.
s	Quand le fichier est détruit, les blocs de données libérés sont remis à 0.
S	Les écritures dans le fichier sont immédiatement effectuées sur le disque. Le fichier est synchrone.
U	Cet attribut permet de récupérer un fichier détruit.

• Les commandes

```
chattr :   Modifier les attributs
lsattr  :   Afficher les attributs
```

La commande chattr permet de modifier les attributs d'un fichier. Sa syntaxe est la suivante:
chattr [-RV] [-v version] [+=Asacdisu] fichier ...

L'option « -R », comme dans de nombreuses commandes, permet de modifier les attributs de toute une arborescence.

L'option « -v » rend la commande bavarde.

L'option « -v version » permet de modifier la version d'un fichier, initialement fixée à 1.

La commande lsattr visualise les attributs d'un fichier. Sa syntaxe est la suivante.

```
lsattr [-Radv ] [fichier. ... ]
```

L'option « -R » permet de visualiser les attributs de tous les fichiers d'une arborescence.

L'option « -a », comme dans la commande ls, permet de visualiser les attributs des fichiers dont le nom commence par « . ».

L'option « -d », comme dans la commande ls, permet de visualiser les attributs des répertoires et pas leur contenu.

L'option « -v » affiche la version des fichiers.

- Exemples

```
$ls
```

```
f1      f2      f3
```

```
$ chmod +Ss f2
```

```
$ su
```

```
password:
```

```
# chmod +i f1
```

```
# exit
```

```
$ lsattr
```

```
----i---      ./f1
```

```
s--S----- ./f2
```

```
-----      ./f3
```

```
$ rm f1
```

```
rm: détruire le fichier protégé en écriture 'f1'? rm: o
```

```
Ne peut délier 'f1': Opération non permise
```

```
$ chmod -s f2
```

```
$ lsattr f2
```

```
---S----- f2
```


3- Enoncé du TP

TP n°2: Gestion des fichiers

Objectifs:

- Savoir changer les caractéristiques d'un fichier
- Gérer une arborescence de fichiers et de répertoires

Exercice 1:

Recherchez les répertoires de nom « cron » sur votre système.

Exercice 2:

Recherchez tous les fichiers tubes de votre système.

Exercice 3:

Recherchez tous les répertoires accessibles en écriture pour les autres.

Exercice 4:

Connectez-vous avec le compte user01 et créez une arborescence de fichiers en utilisant les commandes suivantes :

```
$ cp /etc/passwd /etc/group -
```

```
$ mkdir ~/boot
```

```
$ cp /etc/inittab /etc/profile ~/boot
```

Listez cette arborescence en utilisant différentes commandes.

Exercice 5:

En étant connecté en tant qu'administrateur, mettez les droits 700 à l'ensemble des fichiers de l'utilisateur user01.

Exercice 6:

En utilisant le manuel, retrouvez les différentes utilisations du droit SGID dans le système Linux.

Exercice 7:

Créez un fichier par la commande cp et rendez-le non modifiable. Listez ses attributs. Essayez de le modifier.