Correction TP N°2 : Système de fichiers

Analyse des droits

Création de l'arborescence

```
Créer les groupes buro et tp
groupadd -g 503 buro
groupadd -g 504 tp

changer le groupe de l'user user1

vi /etc/passwd
user1:x:500:503:user1:/home/user1:/bin/bash

changer le groupe des répertoires existants
chgrp -R 503 user1

créer util1 et util2
useradd -g 504 -u 504 -m util1
useradd -g 504 -u 505 -m util2

changer les mots de passes : passwd util1...

ls /home
drwxr-xr-x  3 user1   user1   4096 oct  2 11:12 user1/
drwxr-xr-x  3 util1   tp      4096 oct  2 10:59 util1/
drwxr-xr-x  3 util2   tp      4096 oct  2 11:00 util2/

ouvrir 3 consoles pour les trois users : Ctrl+Alt+FX
```

user1

```
>tata
-rw-r--r--  1 user1 buro    0 oct  2 11:11 tata
```

util1

```
>toto
chmod o-r toto ou chmod 0620 toto
-rw-r-----  1 util1 tp    0 oct  2 11:21 toto

echo 'echo Bonjour'>modiftoto
chmod 6755 modiftoto ou chmod a+x,ug+s modiftoto
-rwsr-sr-x  1 util1 tp    0 oct  2 11:25 modiftoto*
```

util2

```
>titi
chmod g+w titi
-rw-rw-r--  1 util2 tp    0 oct  2 11:30 titi
```

Analyser les droits d'accès sur les commandes suivantes :

user1 situé dans son répertoire user1 entre :

```
cat /home/util1/toto : pas le droit r pour other
../util1/modiftoto : ok
cd ../util2 : pas le droit x pour other
cp tata ../util1 : pas le droit w sur le répertoire
```

util1 situé dans son répertoire util1 entre :

```
rm ../util2/titi : pas la droit w sur util2
```

util2 situé dans son répertoire util2 entre :

```
rm ../util1/toto : ok
```

Maîtrise des droits d'accès

- Positionnez les droits des répertoires à la création à `rwxrwxrwx`

```
umask 0000
```

- Créez la sous-arborescence suivante à partir de votre répertoire de login ***util***

```
su – util1
mkdir rep1
mkdir rep1/rep2
mkdir rep1/rep3
mkdir rep1/rep3/rep4
```

- Modifiez les droits d'accès des répertoires de manière à ce que :

1. Les utilisateurs n'appartenant pas à votre groupe ne puissent accéder à la sous-arborescence rep1

```
chmod o-x rep1
```

2. rep2 soit pour vous un répertoire privé

```
chmod go-rwx rep1/rep2 ou chmod 0700 rep1/rep2
```

3. rep3 soit partagé en lecture seule pour les utilisateurs de votre groupe

```
chmod g=r-x rep1/rep3
```

4. rep4 soit partagé en lecture et en écriture pour les utilisateurs de votre groupe, inaccessible aux autres

```
chmod g=rwx,o= rep1/rep3/rep4
```

- Créez des fichiers ayant pour droits à la création `rw-r-----` dans les différents répertoires

```
umask 0026
touch fichiers dans les repertoires
umask 0000

ou

touch fichiers dans les repertoires
chmod 0640 fichiers , chmod u=rw,g=4,o= fichiers
```

- Vérifiez le "bon comportement" des droits en essayant de copier, créer, supprimer, lire, modifier fichiers et répertoires. Créez de nouveaux comptes utilisateurs si besoin.

```
…
```

- Vérifiez le mécanisme d'affectation des droits à la création d'un fichier

```
création :
   umask 0000
   touch f ou >f
   ls –l => rw-rw-rw-

création :
   umask 0022
   touch g ou >g
   ls –l => rw-r—r—

copie :
   chmod 0444 f
   cp f g
   ls –l => r--r--r—

vi h :
   ls –l => rw-r—r— (fn du umask)
```

- Vérifiez pour un répertoire le sens des droits :

1. `--x`

```
traversée : cd repertoire
```

2. `r–`

```
lecture : lister : ls repertoire
```

3. `-w-`

```
écriture : créer un fichier touch fichier
```

- Vérifiez le fonctionnement du bit **t** appliqué sur un répertoire

```
Il faut être other pour pouvoir tester la protection
Appliquer le droit t sur rep1
chmod +t rep1 ou chmod 1777 rep1
Créer un fichier appartenant à util2
su util2
touch u2
^D
su user1
rm u2 : impossible
su util1
rm u2 : ok (n'est pas other pour rep)
```

- Positionnez l'attribut **i** sur un fichier (sous root). Vérifiez que dès lors ce fichier ne peut être ni renommé, ni modifié, ni supprimé quel que soit son mode d'accès

```
su
chattr +i u2
rm u2 => impossible
chattr –i u2
rm u2 => ok
```

Les liens physiques

Créez un fichier texte nommé cible dans rep1.

```
su – util1
cd rep1
echo Bonjour>cible
```

Dans chacun des sous-répertoires, créez des liens physiques vers le fichier cible.

```
ln cible rep2
ln cible rep3
ln cible rep3/rep4
```

Utilisez la commande `ls -li` sur chacun de ces liens, vérifiez que chacun dispose du même numéro d'inode et du même nombre de liens.

```
ls –li cible
ls –li rep2
ls –li rep3
ls –li rep3/rep4
(nb liens =4)
```

Supprimer un lien et constatez la décrémentation du nombre de liens.

```
rm rep3/rep4/cible
(nb liens =3)
```

Changez les droits d'accès d'un lien. Que deviennent les droits d'accès des autres liens ?

```
chmod o-r cible
=> changement pour tous (même inode)
```

Créez un lien vers un fichier ne vous appartenant pas. Pouvez vous modifier les droits du lien créé ? Expliquez.

```
ln ~user1/tata
=> non le fichier ne nous appartient pas
```

Vérifiez que l'on ne peut pas créer un lien physique inter-partitions.

```
ln /etc/passwd .
```

Les liens symboliques

Créez un fichier texte nommé cibleSymbolique dans rep1.

```
echo Lien Symbolique>cibleSymbolique
```

Dans chacun des sous-répertoires, créez des liens symboliques vers le fichier cibleSymbolique.

```
cd rep 2; ln –s ../cibleSymbolique lien
cd ../rep3;ln –s ../cibleSymbolique lien
cd rep3/rep4;ln –s ../../cibleSymbolique lien
```

Utilisez la commande `ls -l` sur les liens symboliques.

```
la taille du fichier est fonction du nombre de caractères du chemin
```

Quel est le mode d'accès d'un lien symbolique ? Modifiez ce mode d'accès, interprétez.

```
lrwxrwxrwx (tous les droits)
Modifie les droits du fichier pointé par le lien
```

Supprimez un lien symbolique, interprétez.

```
rm lien
aucune incidence sur le fichier pointé
```

Supprimez le fichier cibleSymbolique. Que deviennent les liens symboliques ?

```
aucune incidence ils pointent sur un fichier n'existant pas
```

Créez un lien symbolique vers un répertoire, testez.

```
ln –s rep3/rep4 test
cd test
pwd /home/util1/rep1/rep3/rep4
rmq : cd .. retourne à rep1!!
```

Vérifiez que l'on peut créer un lien symbolique inter-partitions.

```
ln –s /etc/passwd .
```

Analyser le comportement des commandes suivantes sur les liens symboliques

a. cp :nouvelle inode copie des données
b. mv : copie le lien sans mise à jour di changement de répertoire
c. rm : supprime le lien (pas les données)
d. ln : créer un nouveau lien symbolique de même inode
e. chmod : change les droits du fichier pointé
f. chown : change le propriétaire du fichier pointé
g. cat : liste le contenu du fichier pointé

La commande find

Dans la sous-arborescence précédente, recherchez tous les fichiers dont le nom commence par la lettre a.

```
find –name "a*"
```

Recherchez tous les liens physiques ayant un même numéro d'inode.

```
find –inum xxxx
```

Effacez tous les liens physiques ayant un même numéro d'inode.

```
find –inum xxxx –exec rm {} \;
```

1. Retrouver (tous) :
    a. les fichiers de l'utilisateur user1 dans l'arborescence /home
        - find /home –user util1 –type f

    b. les fichiers vides dont le nom commence par un u
        - find / -empty –name "u*" –type f (ou -size 0)

    c. les liens symboliques dans l'arborescence /bin
        - find /bin –type l

    d. les fichiers ayant les droits rwx pour tous (user group & other)
        - find /home -perm 0777 –type f

    e. les fichiers exécutables par tous
        - find /home –perm -0111 –type f

    f. l'arborescence courante
        - find

    g. les fichiers dont le nom comporte un chiffre dans les répertoires de connexion (directement situés sous /home)
        - find /home -maxdepth 2 -name "*[0-9]*" –type f

    h. les fichiers ayant 3 liens physiques
        - find /home –links 3 –type f

    i. les fichiers sous le répertoire courant ayant été modifiés durant les 5 dernières minutes
        - find –mmin -5 –type f

    j. les répertoires commençant par un u dans l'arborescence /home
        - find /home/u* –type d

1. Compter le nombre de lignes de chaque fichier de /home
    - find /home –name "*" –type f –exec wc –l {} \;

2. Mettre à jour la date de dernière modification pour tous les répertoires de /home
    - find /home –name "*" –type d –print –exec touch {} \;

3. Supprimer tous les fichiers n'appartenant pas à un utilisateur déclaré dans /etc/passwd
    - find / –uname –type f -delete

4. Copier tous les fichiers commençant par une étoile de votre arborescence dans le répertoire courant (avec demande de confirmation avant la copie)
    - find ~ –name "**.*" –type f -ok cp {} . \;

5. Pour tous les fichiers appartenant à des membres du groupe 1000, donner comme propriétaire util1
    - su
    - find /home –gid 1000 –type f -exec chown user1 {} \;

Exercices - Corrections

Droits

Création de site web

Une équipe est chargée de créer un site web.

Cette équipe est composée de designers développeurs et commerciaux

Pour structurer leur travail, l'administrateur doit leur fournir un espace de travail dans lequel se trouvent :

- Un répertoire accessible à tous contenant :
  - un fichier 'intouchable' contenant le cahier des charges
  - des fichiers déposés par les développeurs et designers
    - seuls habilités à les créer et les détruire
- Un répertoire dédié aux designers
- Un répertoire dédié aux développeurs
- Un répertoire partagé par les développeurs et designers
  - seuls les propriétaires des fichiers présents peuvent les détruire

Donner la structure de l'arborescence et les commandes permettant de la réaliser

Dans une session root :

Créer un groupe dev et y ajouter les développeurs
Créer un usager admindev
Créer un groupe des et y ajouter les designers
Créer un usager admindes
Créer un groupe devdes de et y ajouter les développeurs et les designers
Créer un usager admindesdev

```
mkdir /home/projet
chattr +i cahier_des_charges
chmod 755 devdes

mkdir /home/projet/devdes
chown admindevdes:devdes /home/projet/devdes


chmod   1770 /home/projet/devdes
sticky bit : seuls les propriétaires peuvent détruire

mkdir /home/projet/dev
chown admindev:admindev /home/projet/dev
chmod 770 /home/projet/dev

mkdir /home/projet/des
chown admindes /home/projet/des
chmod 770 /home/projet/des
```

En utilisant la commande rm, créer dans le répertoire de login de *user1* une commande *rmlibre* permettant à tout utilisateur de détruire les fichiers créés par *user1*.

Avec une session root

Copier l'exécutable dans le répertoire de login

```
cp /bin/rm ~user1/rmlibre
```

Changer le propriétaire du fichier

```
chown user1 ~user1/rmlibre
```

Attribuer le droit s au fichier

```
chmod a+s ~user1/rmlibre
```

Tester en se loggant user2

3. Attributs étendus

a. Créer un fichier immutable et vérifier qu'il est effectivement protégé
b. Tester également l'attribut A
c. Vérifier le fonctionnement lors des copies