

Corrigé- TD/TP N° 2 (Unix/Linux- Protection des fichiers)

Exercice 1

	Notation symbolique	Notation Octale
Commande 1	<code>chmod u=rwx,g=rx,o=x monRep</code>	<code>chmod 751 monRep</code>
Commande 2	<code>chmod u=rx,g=w,o=x monRep</code>	<code>chmod 521 monRep</code>
Commande 3	<code>chmod u=w,g=x,o=r monRep</code>	<code>chmod 214 monRep</code>
Commande 4	<code>chmod u=x,g=rx,o= monRep</code>	<code>chmod 150 monRep</code>

Exercice 2

On suppose que le dossier *monRep* contient déjà le fichier *test* et que ce dernier a les droits "*rw-r--r--*".

- Commande 1 : le propriétaire peut **entrer dans le dossier, voir son contenu et modifier le fichier**.
- Commande 2 : le propriétaire peut **entrer dans le dossier, voir son contenu et modifier le fichier**.
- Commande 3 : Le propriétaire **ne peut ni entrer dans le dossier, ni voir son contenu, ni modifier le fichier**. Pour **accéder** à un fichier dans un répertoire (que ce soit pour lire, modifier ou exécuter le fichier), l'utilisateur doit avoir d'abord avoir le droit d'exécution (x) sur le répertoire contenant le fichier.
- Commande 4 : le propriétaire peut **entrer dans le dossier** mais **ne peut ni voir son contenu ni modifier le fichier**.

Conditions pour **lire** un **fichier** dans un répertoire :

- Droit de lecture (r) sur le fichier lui-même ;
- Droit d'exécution (x) sur le répertoire contenant le fichier.

Conditions pour **modifier** le contenu d'un **fichier** dans un répertoire :

- Droit d'écriture (w) sur le fichier ;
- Droit d'exécution (x) sur le répertoire contenant le fichier

Exercice 3

- 1- Créer et remplir le fichier *bienvenue* au sein du répertoire *repTest* ;
- 2- Ajouter le droit d'exécution "x" au fichier *bienvenue* : `chmod u+x repTest/bienvenue`
- 3- Exécuter le script *bienvenue* : `./repTest/bienvenue`

Remarque : On peut ajouter le chemin "*~/repTest*" à la variable d'environnement *PATH* (*PATH=\$PATH:/home/ahmadi/repTest*). Dans ce cas, pour l'exécution, il suffit de taper : *bienvenue*

Exercice 4

Pour que le propriétaire puisse **créer** ou **supprimer** un fichier au sein d'un répertoire il doit avoir le droit "w" sur ce répertoire.

1. `touch monFichier.txt` # Créer le fichier
2. `chmod 600 monFichier.txt` # Définir les permissions
3. `ls -l monFichier.txt` # Vérifier les permissions
4. `cat monFichier.txt` # Lire le fichier
5. `nano monFichier.txt` # Modifier le fichier
6. `rm monFichier.txt` # Supprimer le fichier

Exercice 5

1. `mkdir monRep` # Créer un répertoire
2. `chmod 300 monRep` # Définir les permissions du répertoire (-wx --- ---)
3. `touch monRep/monFichier.txt` # Créer un fichier
4. `chmod 400 monRep/monFichier.txt` # Définir les permissions du fichier
5. `ls -l monRep/monFichier.txt` # Vérifier les permissions
6. `cat monRep/monFichier.txt` # Lire le fichier
7. `rm monRep/monFichier.txt` # Supprimer le fichier

Exercice 6

1. `mkdir dossEssai`
2. `cat > dossEssai/monFich`
*Ceci est un fichier
que le propriétaire peut lire,
mais ne peut ni modifier
ni supprimer
<Ctrl-D>*
3. `chmod 400 dossEssai/monFich` # ou `chmod 444 dossEssai/monFich`
4. `chmod -w dossEssai/`
5. `rm dossEssai/monFich`
*rm : supprimer 'dossEssai/monFich' qui est protégé en écriture et est du
type « fichier » ? o
rm: impossible de supprimer 'dossEssai/monFich': Permission non accordée*
6. `cat >> dossEssai/monFich` # tentative d'ajouter du texte en fin du fichier
bash: dossEssai/monFich: Permission non accordée

Exercice 7

Quand le nouveau fichier est créé par la commande "**cp**":

`cp fich1 fich1Bis` : *fich1Bis* prendra les permissions du *fich1* et non pas celles définies par **umask**.

Remarque :

La permission "**x**" n'est jamais attribuée à un fichier à sa création, même si umask le spécifie.

Exercice 8

1. **Création d'un groupe commun** : **ali** doit d'abord créer un groupe qui comprendra à la fois **ali** et **nadia**. Ce groupe sera utilisé pour gérer les accès au répertoire partagé.

```
sudo groupadd collaborateurs
```

```
sudo usermod -aG collaborateurs ali,nadia
```

L'option **-aG** ajoute les utilisateurs sans les retirer des autres groupes.

2. **Création du répertoire partagé** : **ali** va maintenant créer un répertoire appelé **projet** dans le dossier **/usr/local/share** où les fichiers partagés seront stockés.

```
sudo mkdir /usr/local/share/projet
```

3. **Vérification des permissions initiales** : Après la création du répertoire, vérifier ses permissions :

```
ls -ld /usr/local/share/projet
```

On devrait obtenir un résultat similaire à celui-ci :

```
drwxr-xr-x 2 root root 4096 2023-10-07 /usr/local/share/projet
```

4. **Modification du groupe propriétaire du répertoire** : **ali** doit maintenant modifier les permissions du répertoire pour qu'il appartienne au groupe **collaborateurs**. Cela permettra à **ali** et **nadia** d'y accéder.

```
sudo chown :collaborateurs /usr/local/share/projet
```

Ensuite, modifier les permissions du répertoire pour permettre l'écriture au groupe :

```
sudo chmod 775 /usr/local/share/projet
```

Vérifier les nouvelles permissions avec :

```
ls -ld /usr/local/share/projet
```

5. **Problème de création de fichiers** : Après avoir configuré les permissions de base, **ali** crée un fichier **test** dans le répertoire **projet** :

```
> /usr/local/share/projet/test_fichier
```

Vérifier les permissions du fichier créé :

```
ls -l /usr/local/share/projet
```

On devrait voir que le fichier appartient à **ali** et non au groupe **collaborateurs**. **nadia** ne pourra donc pas modifier ce fichier.

6. **Résolution des permissions par défaut** : Pour résoudre ce problème, deux solutions sont proposées :
 - Changer l'**umask** : **ali** et **nadia** doivent changer leur **umask** à **0002** pour que les fichiers créés soient modifiables par les membres du groupe.

```
umask 0002
```
 - Configurer le bit **SetGID** sur le répertoire : **ali** doit modifier les permissions du répertoire **projet** pour que tous les fichiers créés dans ce répertoire appartiennent au groupe **collaborateurs** :

```
sudo chmod g+s /usr/local/share/projet
```

7. **Vérification finale** : Après avoir défini le **bit SetGID**, créer à nouveau un fichier et un sous-répertoire dans **projet** pour vérifier que les permissions sont bien configurées :

```
> /usr/local/share/projet/test_fichier
```

```
mkdir /usr/local/share/projet/test_dossier
```

```
ls -l /usr/local/share/projet
```

On devrait voir que les fichiers et répertoires créés appartiennent au groupe **collaborateurs** et sont modifiables par **ali** et **nadia**.