Laboratoire 4 – Linux - corrigé

2.3 Permissions des répertoires

Passer les commandes **`cd /`** puis **`ls -lLd *`**, pour lister les répertoires situés à la racine.

A qui appartiennent les fichiers dans la racine du système de fichiers ?

Les répertoires dans la racine Linux appartiennent à l'utilisateur et au group **`root`**. On peut noter **`root:root`**.

Un utilisateur quelconque peut-il y créer des sous-répertoire ?

Non, les droits en écriture ne sont donnés qu'au propriétaire : **`drwxr-xr-x`**.

Commentez les 2 cas particuliers **`/root`**, **`/lost+found`** et **`/tmp`**.

Seul **`/root`** et **`/lost+found`**, sont avec des accès restreints uniquement pour le propriétaire **`drwx------`**. **`/tmp`** est par contre accessible à tous en lecture et écriture **`drwxrwxrwt`**.

2.5 Exercice 1 – Les droits d'accès

1. Quels sont les droits sur votre répertoire personnel (pour rappel **`cd /home`** pour se déplacer dans le dossier parent de votre home directory) ? La propriété du dossier est **`MonLogin:student`** et **`rwx`** pour le moi-même, **`r-x`** pour le groupe **`student`** et **`r-x`** pour les autres.
2. Un utilisateur différent hacker peut-il y pénétrer ou seulement lister vos fichiers ? Expliquez pourquoi. Oui, puisqu'il est membre du groupe **`student`** et que l'accès **`r-x`** autorise de parcourir le contenu du dossier.
3. Pour tester vos affirmations, vous pouvez changer d'utilisateur dans votre shell via la commande **`sudo su – hacker`** et essayer d'accéder à vos données. Attention, ne restez pas connecté en tant que hacker sur votre shell, tapez **`exit`** !
4. Et l'utilisateur louis, le pourrait-il ? Sachant qu'il ne fait pas partie du groupe **`student`** ? Expliquez pourquoi. Oui, puisque la catégorie des autres utilisateurs (other) possède aussi l'accès **`r-x`**, ce qui autorise de parcourir le contenu du dossier.

3.4 Exercice 2 – manipulations des droits d'accès

1. Exécutez les commandes pour changer les droits sur votre dossier personnel pour retirer tous droits d'accès à vos données à l'utilisateur **`louis`**.

```
chmod o-rx ~
```

2. Donnez les trois manières d'écrire la commande **`chmod`** pour réaliser l'opération précédente.

```
chmod o-rx ~
chmod o= ~
chmod 750 ~
```

3. Sans utiliser le compte **root**, faite de même pour appliquer les mêmes droits sur le home directory de l'utilisateur **hacker**.

```
sudo su - hacker
chmod o-rx ~     # une des trois commandes
chmod o= ~
chmod 750 ~
```

4. Comparer les permissions de **/etc/passwd** et **/etc/shadow**. Pourquoi a-t-on nommé ainsi ce dernier fichier ?

```
ll /etc/passwd /etc/shadow
-rw-r--r-- 1 root root   3311 Sep 24 18:16 /etc/passwd
-rw-r----- 1 root shadow 1966 Sep 24 18:15 /etc/shadow
```

Les données utilisateurs contenues dans **/etc/passwd** sont accessibles en lecture à tous les utilisateurs du système. Seul **root** peut y écrire et donc ajouter/modifier des utilisateurs. **/etc/shadow**, car il s'agit d'un fichier sensible qui contient les mots de passe « hashés » des utilisateurs locaux au système Linux. Ces informations doivent donc rester dans cachées, seul le root et le groupe **shadow** possède l'accès à ce fichier. Sur certains système Linux, les droits sur /etc/shadow sont **r--------**

5. **hacker** peut-il lire le fichier **/etc/shadow** ?
   Non.
6. Visualiser sa présence dans le dossier **/etc** ?
   ```
   ll /etc/shadow
   ```
7. Examiner son contenu ?
   Impossible sans passer **root**.
   ```
   sudo su –
   less /etc/shadow
   exit
   ```
8. A partir de votre compte utilisateur, essayez de faire une copie de **/etc/shadow** sous le nom **~/shadow.bak** ! Vérifiez les droits sur la copie du fichier et concluez !
   ```
   cp /etc/shadow ~/shadow.bak
   cp: cannot open '/etc/shadow' for reading: Permission denied
   ```
9. Toujours à partir de votre compte utilisateur, essayez de faire une copie de **/etc/passwd** sous le nom **~/passwd.bak** ! Vérifiez les droits sur la copie du fichier et concluez !
   ```
   cp /etc/passwd ~/passwd.bak
   ll ~/passwd.bak
   -rw-r--r-- 1 p200010 student 3311 Oct 11 07:37 /home/p200010/passwd.bak
   less ~/passwd.bak          # la touche Q pour quitter l'affichage du contenu
   ```
10. Dans le fichier **~/passwd.bak**, supprimez la ligne de l'utilisateur **louis** soit avec l'éditeur **nano** ou **vi** si vous le maitrisez.
    ```
    nano ~/passwd.bak          # CTRL+X pour quitter
    ```
11. Essayez de faire la copie inverse **~/passwd.bak** vers **/etc/passwd**. Concluez !
    ```
    cp ~/passwd.bak /etc/passwd
    cp: cannot create regular file '/etc/passwd': Permission denied
    ```

12. En vous connectant sous le compte **root** faite maintenant une copie de **/etc/shadow** dans votre home directory, sous le nom **/home/*[VOTRE_COMPTE]*/shadow.bak**

```
sudo su –
cp /etc/shadow /home/VOTRE_COMPTE/shadow.bak
```

13. Déplacez-vous dans le dossier **/home/*[VOTRE_COMPTE]*** et accordez-vous la propriété de la copie. Comment réalisez-vous ces opérations ?

```
cd /home/VOTRE_COMPTE
chown VOTRE_COMPTE shadow.bak
ls -l shadow.bak
-rw-r----- 1 VOTRE_COMPTE root 1966 Oct 11 07:51 shadow.bak
```

14. Revenez à votre compte utilisateur et vérifiez si vous avez l'accès au contenu du fichier **~/shadow.bak** en modifiant quelques lignes de son contenu.

```
exit
nano shadow.bak  # Vous avez accès au contenu en lecture écriture (CTRL+X pour quitter)
```

15. Supprimez cette copie **~/shadow.bak**, car celle-ci contient des données sensibles et représente une faille sécurité sur votre système d'exploitation.

```
rm ~/shadow.bak
```

16. Avec l'utilisateur **hacker** ou votre propre utilisateur, pouvez-vous créer le répertoire temporaire **/home/temp** ? essayez ! pourquoi ?

```
sudo su - hacker
mkdir /home/temp
mkdir: cannot create directory ‘/home/temp’: Permission denied
ls -ld /home
drwxr-xr-x 5 root root 4096 Sep 24 18:14 /home
```

Seul **root** possède l'accès en écriture sur le dossier **/home**

Effectuez cette création comme **root**.

```
exit        # n'oubliez pas de sortir de session hacker
sudo su –
mkdir /home/temp
```

17. Accordez les permissions maximales sur **/home/temp** et vérifiez.

```
chmod 777 /home/temp        # une des trois commandes
chmod a=rwx /home/temp
chmod ugo=rwx /home/temp
exit        # ATTETION, NE RESTEZ PAS CONNECTE EN ROOT
```

18. Avec l'utilisateur **hacker**, tout content d'avoir enfin un droit d'écriture dans **/home/temp** essayez de copier les 2 fichiers système **/etc/hosts** et **/etc/passwd** dans **/home/temp**.

Avez-vous les droits suffisants pour le faire ? Oui

Pourquoi ? Parce que le fichier /etc/passwd et /etc/hosts autorise l'accès en lecture sur la catégorie other.

Qu'affiche la commande **ll /home/temp** ?

```
sudo su - hacker
cp /etc/hosts /home/temp
cp /etc/passwd /home/temp
ll /home/temp
total 8
```

```
-rw-r--r-- 1 hacker student  211 Oct 11 11:50 hosts
-rw-r--r-- 1 hacker student 3311 Oct 11 11:51 passwd
```

19. L'utilisateur hacker doit vous donner les accès sur ces deux copies dans **/home/temp**. Mais il veut retirer ses propres accès à ces deux fichiers. Comment s'y prend-t-il ? Réalisez l'opération.
    Il faudrait changer le propriétaire des deux fichiers.
    chown p200010 ./hosts ./passwd
    chown: changing ownership of './hosts': Operation not permitted
    chown: changing ownership of './passwd': Operation not permitted
    Ceci est impossible, car vous ne pouvez changer le propriétaire d'un fichier qui vous appartient sans être **root** !

4.1 Exercice 3 – droits d'accès par défaut

1. Donnez le **umask** de votre utilisateur sous forme octal.
   umask
   0022
2. Que seront les droits d'accès par défaut lors de la création d'un fichier avec ce **umask** ?
   rw-r--r--
   Expliquez votre réponse.
   0666 → droit maximum pour un fichier
   -0022
   =0644 → **rw-r--r--**
   Que seront les droits d'accès par défaut lors de la création d'un dossier avec ce **umask** ?
   rwxr-xr-x
   Expliquez votre réponse.
   0777 → droit maximum pour un dossier
   -0022
   =0755 → **rwxr-xr-x**