

Compte rendu

Gestion des utilisateurs

Introduction : Le but de ce TP, est de comprendre est de créer, géré, modifié des comptes utilisateurs sur un système Linux (système multi-utilisateurs). Ce TP permet également de correctement appréhender les droits spéciaux ainsi que de pouvoir apprendre à gérer des groupes. Une identification est nécessaire pour leur permettre de se connecter sur leur session respective et ainsi gérer leurs courriers électroniques par exemple.

Un utilisateur a besoin essentiellement d'un : nom d'utilisateur, un mot de passe, un groupe, un identifiant, un interpréteur de commandes et d'un répertoire personnel.

I) Son contenu :

1) Ajout d'un utilisateur

Dans ce première partie, il s'agit de créer un nouvel utilisateur « astérix » en lui attribuant un login, un mot de passe afin qu'il puisse se connecter au système.

```
root@debND:~# useradd -m asterix
root@debND:~#
```

On commence par ajouter un nouvel utilisateur avec la commande `useradd -m asterix`

Question 1 : Le paramètre `m` permet de créer un répertoire personnel pour le nouvel utilisateur s'il n'existe pas (la même chose que la commande `-create-home`).

Question 2 : Le répertoire personnel de cet utilisateur se situe dans `/home/asterix:/bin/sh`.

```
root@debND:/etc# cat passwd | grep asterix
asterix:x:1001:1001:~/home/asterix:/bin/sh
```

Question 3 : Oui, il est indispensable de créer un répertoire personnel d'un utilisateur car cela permet de stocker ses fichiers personnels et les sous-répertoires et ainsi de savoir quel utilisateur il s'agit.

Question 4 : Oui, il possède un **shell** afin de créer ce **shell** il faut entrée la commande **cat /etc/passwd** qui permet tout simplement de visualiser les informations sur l'utilisateur.

Question 5 : Oui, car l'utilisateur a forcément toujours besoin d'un shell sinon il ne pourrait plus se connecter sur son compte personnel.

Question 6 : **adduser** est une commande interactive qui permet de créer un nouvel utilisateur en posant des questions comme dans un formulaire.

Useradd est une commande non interactive qui permet elle aussi de créer un nouvel utilisateur.

Question 7 : Pour définir un mot de passe il faut saisir la commande **passwd** asterix

```
Entrez le nouveau mot de passe UNIX :
Retapez le nouveau mot de passe UNIX :
passwd : le mot de passe a été mis à jour avec succès
```

Le numéro d'UID de l'utilisateur asterix est donné en utilisant la commande **cat etc/passwd**.

```
sio1a:x:1000:1000:sio1a,,,:/home/sio1a:/bin/bash
asterix:x:1001:1001:~/home/asterix:/bin/sh
root@debND:/etc# _
```

Question 8 : Pour créer ces deux nouveaux user, il faut entrer les commande suivantes.

```
root@debND:/etc# useradd -m obelix
root@debND:/etc# useradd -m Panoramix
```

Question 9 : L'option **-r** dans la commande précédente permet de supprimer les fichiers personnels de l'utilisateur et sa messagerie (**-r** signifie remove).

2) Modification d'un compte

Dans cette deuxième parties, il s'agit de pouvoir modifier un compte utilisateur, changer son login son répertoire personnel crée par défaut, de verrouiller et de déverrouiller son mot de passe etc.

Question 1 : Le paramètre qui permet de changer le nom de login d'un utilisateur est « **-l** »

```
-l, --login IDENTIFIANT    définir un nouveau nom pour le compte
```

Question 2 : Le paramètre qui permet de changer le répertoire personnel est « **-d** » comme vue précédemment.

```
-m, --move-home            déplacer le contenu du répertoire personnel
                           vers le nouvel emplacement (à n'utiliser
                           qu'avec -d)
```

Question 3 : Le paramètre qui permet de changer le répertoire personnel de l'utilisateur en déplaçant le contenu de l'ancien vers le nouveau est « **-m** » à utiliser qu'avec le **-d** soit la commande : **-m -d move-home**.

```
-m, --move-home            déplacer le contenu du répertoire personnel
```

Question 4 : Le paramètre qui permet de verrouillé le mot de passe d'un utilisateur est « **-L** »

Et le paramètre pour déverrouillé est « **-U** » soit la commande : **-l -lock** et **-U -unlock**.

```
-L, --lock                 bloquer le compte de l'utilisateur
```

3) Les groupes

Dans ces troisièmes parties, il s'agit de pouvoir créer un groupe d'utilisateur qui s'appelle **Gaulois** qui permet de partager les mêmes fichiers et les mêmes répertoires. Nous allons ainsi créer de nouveaux utilisateurs tels que **Stagey** et **Totox** et une fois le groupe **Gaulois** créé, nous ajouterons ces deux nouveaux utilisateurs à ce groupe.

Question 1 : Pour créer deux nouveaux utilisateurs, on effectue les commandes suivantes (avec Shell

```
root@debND:~# useradd -m stagey
root@debND:~# useradd -m totox
```

et le répertoire personnel) suivantes :

Question 2 : Voici le compte « **Stagey** »

```
debian login: stagey
Password:
Linux debian 3.2.0-4-686-pae #1 SMP Debian 3.2.63-2+deb7u1 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
$ _
```

Et voici le compte « **Totox** »

```
debian login: totox
Password:
Linux debian 3.2.0-4-686-pae #1 SMP Debian 3.2.63-2+deb7u1 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
$ _
```

Question 3 : Si on crée déjà un compte existant, la machine nous répond qu'il en existe déjà et il se trouve dans le répertoire /home.

```
root@debian:~# useradd totox
useradd : l'utilisateur « totox » existe déjà
```

Question 4 :

```
root@debian:~# userdel -r totox
userdel : l'emplacement de boîte aux lettres de totox (/var/mail/totox) n'a pas
été trouvé
root@debian:~# cd /home
root@debian:/home# ls -l
total 36
drwxr-xr-x 2 asterix asterix 4096 nov. 6 15:14 asterix
drwx----- 2 root root 16384 nov. 6 14:56 lost+found
drwxr-xr-x 2 obelix obelix 4096 nov. 6 17:12 obelix
drwxr-xr-x 2 panoramix panoramix 4096 nov. 6 17:19 panoramix
drwxr-xr-x 2 stagey stagey 4096 nov. 13 14:13 stagey
```

On remarque bien que « Totox » à correctement été supprimé et son répertoire personnel aussi.

Question 5 : Le groupe **Gaulois** a été correctement créé.

Question 6 : Les utilisateurs asterix, obelix ont bien été ajoutés dans le groupe « Gaulois ».

```
root@debian:/# gpasswd -a asterix gaulois
Ajout de l'utilisateur asterix au groupe gaulois
root@debian:/# gpasswd -a obelix gaulois
Ajout de l'utilisateur obelix au groupe gaulois
```

Question 7 : On remarque bien que l'utilisateur obelix appartient au groupe : obelix gaulois.

```
debian login: obelix
Password:
Last login: Thu Nov 13 14:55:36 CET 2014 on tty1
Linux debian 3.2.0-4-686-pae #1 SMP Debian 3.2.63-2+deb7u1 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
$ groups
obelix gaulois
$ _
```

Question 8 : La commande groups permet de pouvoir lister tous les groupes d'un utilisateur auxquels il appartient.

Question 9 : Pour voir tous les répertoires personnel il faut aller dans /home.

Question 10 : Le caractère X signifie que les mots de passes des utilisateurs sont cryptés.

Question 11 : Voici le screen pour la description d'un utilisateur.

```
root@debian:/# chfn obelix
Modification des informations relatives à l'utilisateur obelix
Entrez la nouvelle valeur ou « Entrée » pour conserver la valeur proposée
  Nom complet []: obelix
  N° de bureau []: prof
  Téléphone professionnel []: 0299999999
  Téléphone personnel []: 0660070809
  Autre []:
```

4) Les commandes de recherches

Dans ces quatrièmes parties, il s'agit de savoir notre numéro GID (numéro d'identifiant de groupe auquel appartient l'utilisateur), il s'agira de rechercher tous les fichiers de périphériques de l'utilisateur.

Question 1 : Mon numéro GID est 1000.

Le numéro GID est 0.

```
root@debian:/# cat /etc/passwd |grep root
root:x:0:0:root:/root:/bin/bash
```

Question 2 : En tapant la commande « **/find/home/nathan** », cela permet de visualiser l'arborescence de tous les fichiers qui m'appartient.

Question 3 : Afin de rechercher tous les fichiers non accèdes depuis 3 jours par exemple, il faut utiliser la commande suivante : « **find** » qui permet de rechercher tous les fichiers.

Question 4 : Afin de rechercher les fichiers de périphériques, il faut utiliser la commande suivante « **cat /proc/devices** ».

```
252 hidraw
253 bsg
254 rtc

Block devices:
259 blkext
  7 loop
  8 sd
 11 sr
 65 sd
 66 sd
 67 sd
 68 sd
 69 sd
 70 sd
 71 sd
128 sd
129 sd
130 sd
131 sd
132 sd
133 sd
134 sd
135 sd
```

Question 5 : Pour que cette commande puisse ignorer la casse des caractères dans sa recherche, que ce soit en majuscule et en minuscules, il suffit d'utiliser les outils suivants : « **sed et cut** », Ils permettent de modifier de supprimer ou de remplacer des occurrences dans une chaîne de caractères sachant qu'une occurrence définit un exemplaire de données. De plus, pour que la commande **grep** ignore la casse des caractères dans sa recherche, il faut utiliser la commande **find**. Pour ce faire, le paramètre **-iname** doit être mis en valeur.

Question 6 : La commande **locate** est forcément nécessaire pour utiliser une base de données de l'arborescence des fichiers personnels de l'utilisateur. La commande ping6 se trouve dans le fichier binaire **/bin** c'est-à-dire **/bin/ping6**.

5) Les droits spéciaux

Dans cette cinquième partie, il s'agira de savoir les droits attribués aux utilisateurs créés.

Question 1 : Le programme **passwd** est accessible dans : « **/etc/passwd** » qui permet de visualiser toutes les informations des comptes des utilisateurs.

Les droits affectés au programme **passwd** sont :

- **r** (read : lire) permet de lire le fichier.
- **w** (write : écrire) permet d'écrire et de modifier le fichier.

L'utilisateur ne possède pas le droit d'exécution du fichier (x : exécuter) car seul le propriétaire de ce fichier possède ce droit d'exécution.

Question 2 : Il peut devenir setuid, cela veut dire que lorsqu'un fichier est exécuté il détient les droits de son propriétaire. Le programme **passwd** permet de modifier son MDP est setuid root.

Question 3 : Le **sticky bit** signifie que cela reste en mémoire a la fin de son exécution afin de pouvoir être relancé plus rapidement. Par contre, seulement l'administrateur du système peut s'occuper du **sticky bit**.

Question 4 : Dans le répertoire créé /tmp, Tous les utilisateurs ont ainsi le droits de mettre des informations. Le **sticky bit** doit être absolument être mis en valeur.

6) Le partage de répertoire et des permissions

Dans cette sixième partie, il s'agira de modifier les permissions attribuées à chaque utilisateur pour ainsi leur permettre d'avoir accès aux répertoires des autres utilisateurs qui sont eux-mêmes intégré dans le même groupe.

Question 1 : Le répertoire « **village** » a été créé dans le répertoire /home

```
root@debian:/# cd /home
root@debian:/home# mkdir village
root@debian:/home# cd /home/village
root@debian:/home/village# _
```

Question 2 : Afin de pouvoir modifier des permissions dans un répertoire quelconque, il faut absolument mettre en œuvre la commande suivant ; **chmod** avec le nom de l'utilisateur concerné et son chemin de répertoire personnel de cet utilisateur.

Question 3 : Nano est un éditeur de texte qui est déjà installé sur la machine, afin que cela puisse fonctionner au démarrage, il suffit simplement de lancer cet éditeur de texte avec le nom du fichier qui doit être créé sous la forme suivant : nano -w nom fichier.

De plus, le paramètre -w permet d'arrêter le retour à la ligne mais cela peut accidentellement causer des problèmes de lecture des fichiers.

Question 4 : Root est le nom d'utilisateur principal du système, par ce qu'il permet d'autoriser tous les droits aux utilisateurs. Il y a forcément un root sur toutes les applications qui sont installé. Il n'est pas nécessaire de le créer. De toute manière, dès le début il sera installé au démarrage du système.

Le problème ne peut pas venir de là, et qu'il est réglé de manière définitive et assure ainsi une protection du propriétaire sur ces fichiers personnels.

Question 5 : Afin de procéder à une modification des permissions sur le répertoire dit « enfant », il y a qu'une seule et unique commande pouvant être utilisée qui est : **chmod**.

Afin de supprimer un sous-répertoire ou un répertoire lui-même, il faut entrer la commande suivante : **rm -r plus le nom du répertoire**.

Question 6 : La création d'une hiérarchie de répertoire doit se faire avec la commande suivante : **mkdir /travail/cours/S11/linux**.

Question 7 : Les permissions 644 pour le répertoire /travail/cours concernent :

- Le propriétaire avec les droits de lecture et d'écriture des fichiers.
- Le groupe avec seulement le droit de lecture des permissions.
- Les autres avec seulement les autorisations de lecture.

Question 8 : La commande qui permet de supprimer un sous-répertoire est : **rm -r /travail/cours/S11/linux**.

Conclusion : Nous avons vu la façon de créer des comptes utilisateurs et apprendre à pouvoir les gérer, les modifier, voire même les supprimer, nous avons vu également comment verrouiller ou déverrouiller le mot de passe d'un ou plusieurs utilisateurs.

Ensuite, nous avons vu comment faire pour que les utilisateurs puissent être dans un groupe et pour ensuite faire un partage de fichier.

Ensuite, nous avons visualisé l'arborescence de tous les fichiers de périphériques qui appartiennent aux utilisateurs et son identifiant de groupe (GID).

Et pour finir, nous avons également vu et distingué les différents droits pour chaque utilisateur et aussi nous avons appris à créer des répertoires pour tous les utilisateurs appartenant à un groupe d'utilisateur.