SYSTÈMES ET RÉSEAUX

Aix*Marseille université

Licence MIAGE GAP

Configuration et gestion d'un serveur

Labo 6 : Gestion des utilisateurs

Objectifs :

Sécuriser son système Linux en interdisant au compte root de travailler.
Maîtriser les commandes système de gestion des groupes et des utilisateurs.

Plan :

Ressources :

http://www.gap.univ-mrs.fr/m4/
pages de manuel Internet : man page of

1. Et si on se passait de root ?

1.1. Qui va prendre sa place ?

Se passer de root est faisable, mais par n'importe comment. En effet, root est le seul utilisateur à avoir tous les droits, et notamment ceux de gestion des autres utilisateurs, des processus, seul autorisé à installer des paquets et à paramétrer les démons, ...

Avant de se passer de root, il faut transférer ses pouvoirs à un autre utilisateur. Pour l'instant, vous n'en avez qu'un, mais rien n'empêchera d'en faire de même pour ceux que nous allons créer ensuite.

Δ seul root a le pouvoir de transférer ses pouvoirs :

```
# apt-get install sudo
```

sudo permettra d'exécuter toutes les commandes avec les pouvoirs de root !

Il faut maintenant éditer le fichier de configuration pour donner tous les pouvoirs au groupe sudo :

```
# nano /etc/sudoers
```

Vous devez alors vous assurer de voir, à modifier ou créer au besoin, la ligne suivante :

```
# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
```

Maintenant, les membres du groupe sudo ont les pleins pouvoirs.

Il ne reste plus qu'à affecter au moins un utilisateur à ce groupe :

```
# adduser <login> sudo
```

pour vérifier, il faut se loguer avec le compte choisi ci-dessus, et taper une commande réservée à root, comme par exemple :

```
$ sudo apt-get update
```

Les messages suivants indiquent un échec. Si la mise à jour se fait, c'est tout bon.

```
E: Impossible d'ouvrir le fichier verrou /var/lib/apt/lists/lock - open (13: Per
mission non accordée)
E: Impossible de verrouiller le répertoire /var/lib/apt/lists/
E: Impossible d'ouvrir le fichier verrou /var/lib/dpkg/lock - open (13: Permissi
on non accordée)
E: Impossible de verrouiller le répertoire d'administration (/var/lib/dpkg/). Av
ez-vous les privilèges du superutilisateur ?
```

1.2. root devient un fantôme.

root été déjà interdit de connexion via ssh, pour réduire les failles de sécurité.
Mais maintenant que root n'est plus utile, le mieux est de le désactiver. Il sera toujours présents, pourra toujours travailler, mais ne pourra plus se loguer.

Les commandes suivantes ne peuvent fonctionner que si vous avez les super-pouvoirs :

```
$ sudo passwd -l root
```

La commande `passwd` sert habituellement à modifier le mot de passe d'un utilisateur.
L'option `-l` permet d'activer l'option `lock` sur le compte.

Pour réactiver le compte root (on ne sait jamais), il suffira de saisir :

```
$ sudo passwd -u root
```

L'option `-u` permet d'activer l'option `unlock` sur le compte.

A partir de maintenant, il est impossible de se connecter avec le compte root. Il restera toutefois possible de travailler avec ce compte, à condition d'usurper son identité :

```
$ sudo su
```

Vous constatez que le compte actif est bien root...

Au niveau sécurité, c'est déjà bien, mais le risque reste potentiel pour les intrusions via les failles de sécurité. On peut faire bien plus.

1.3. Et si on éliminait root ?

Puisqu'il ne sert plus à rien, le plus simple reste de lui interdire toute commande. Pour faire cela, il faut se rappeler que l'interpréteur de commande est un shell. Par défaut les utilisateurs utilisent celui de base, appelé `shell`...
Mais celui-ci a fait l'objet d'améliorations, pour obtenir d'autres shell, aux commandes plus riches, comme celui que vous utilisez en ce moment, qui est le `bash`. Par exemple, la complétion au clavier, l'historique des commandes, ..., ont été ajoutés.

Il suffit donc de retirer à root le droit d'utiliser un shell et le tour est joué.
Le fichier `/etc/passwd` est celui qui contient les paramètres de base des comptes utilisateurs.
Avant, il contenait aussi les mots de passe, mais le risque était trop grand lorsque les techniques de déchiffrage ont permis de récupérer les mots de passe, stockés dans ce fichier accessible en lecture à tout le monde. Aussi les mots de passe font-ils l'objet d'un fichier, accessible à root seulement : `/etc/shadow`

Modifions le fichier des paramètres de base :

```
$ sudo nano /etc/passwd
```

il faut modifier la ligne :

```
root:x:0:0:root is Dieu:/root:/bin/sh
```

pour obtenir :

```
root:x:0:0:root is Dieu:/root:/bin/false
```

Normalement, même la commande `sudo su` ne fonctionne plus.

2. Gestion des utilisateur.

Pour la gestion de vos sites web, serveur ftp ou tout autre besoin, il peut être intéressant de créer d'autres utilisateurs. Il sera alors possible de gérer au mieux les droits d'accès aux différents dossiers et fichiers du système.

2.1. Les utilisateurs et les groupes.

Tout utilisateur doit appartenir à au moins un groupe, dit groupe principal. Par défaut, la création d'un utilisateur créé automatiquement un groupe du même nom.

La liste des utilisateurs créés dans le système se trouve dans le fichier `/etc/passwd`
La liste des groupes créés dans le système se trouve dans le fichier `/etc/group`.
On y retrouve également la liste des utilisateurs appartenant à chaque groupe.
Vous pouvez vérifier que votre compte appartient d'ailleurs bien au groupe `sudo`.

2.2. Création d'un utilisateur.

La commande de base de création d'un utilisateur est :

```
$ sudo useradd [options] <login>
```

Toutefois, quelques options sont nécessaires :
`-m` permet de préciser qu'il faut créer un répertoire personnel pour l'utilisateur, par défaut dans `/home`
`-d <dossier>` permet de spécifier quel dossier va devenir le dossier personnel de l'utilisateur.
`-s /bin/bash` permet de forcer l'utilisation de `bash` plutôt que `shell` à l'utilisateur.

Ainsi, la commande peut devenir :

```
$ sudo useradd -m -s /bin/bash roland
```

2.3. Création d'un groupe.

Le groupe par défaut associé à un utilisateur lors de sa création, qui porte le même nom est créé automatiquement.
Pour créer un autre groupe, il faut utiliser la commande :

```
$ sudo groupadd <nomGroupe>
```

2.4. Les utilisateurs dans les groupes.

Par défaut, un utilisateur est placé dans son groupe éponyme.
Pour le placer dans un autre groupe, déjà créé, il faut utiliser la commande :

```
$ sudo adduser <login> <nomGroupe>
```

2.5. Modification du mot de passe d'un utilisateur.

Un utilisateur créé comme vu précédemment n'a pas de mot de passe, ce qui lui interdit de se connecter. Pour changer le mot de passe d'un utilisateur, il faut utiliser la commande :

```
$ sudo passwd <login>
```

2.6. Suppression d'un utilisateur ou d'un groupe.

La suppression d'un groupe ne peut se faire que si aucun utilisateur n'est intégré à ce groupe comme groupe primaire ou principal, sous peine de voir apparaître un message d'erreur ci-dessous :

```
groupdel : impossible de supprimer le groupe primaire de l'utilisateur « [illegible] »
```

Pour supprimer un groupe, il faut utiliser la commande :

```
$ sudo groupdel <nomGroupe>
```

Pour supprimer un utilisateur, il faut utiliser la commande :

```
$ sudo userdel -r <login>
```

L'option -r permet de supprimer en même temps son répertoire personnel, son dossier courriel (s'il existe) et son groupe éponyme. Attention, aucune marche arrière n'est possible.

A noter que recréer un groupe ou un utilisateur supprimé par erreur ne lui redonne ni ses anciens documents, ni ses anciens droits, car ce n'est pas le nom qui est utilisé pour les droits, mais l'uid ou le gid, respectivement identifiant de l'utilisateur et identifiant du groupe.

2.7. Exercice.

Exercice :

1. Créer les utilisateurs Paul, Pierre et Jack, avec leur prénom en minuscule comme mot de passe.
2. Créer l'utilisateur Franck, sans mot de passe.
3. Créez le groupe promoAdventure
4. Ajouter les utilisateurs Paul et Pierre dans le groupe promoAdventure.
5. Vérifier la présence de leur répertoire personnel.
6. Tentez une connexion sur l'utilisateur franck (`su franck`)
7. Supprimer l'utilisateur Franck.
8. Vérifier dans les fichiers `passwd`, `group` et dans `/home` si Franck a bien été supprimé.