N. Ferry, S. Lavirotte, D. Lopez, H. Renard
Polytech'Nice – Sophia
PeiP 1
2019-2020

TD séance n° 6
Utilisateurs et Groupes sous Unix et Windows

1 Gestion des utilisateurs Unix

1.1 Super-utilisateur

Linux est un système multiutilisateurs (plusieurs utilisateurs peuvent travailler sur la même machine) et sécurisé. Chaque utilisateur, pour travailler, doit s'identifier. Ses fichiers lui appartiennent et il peut en autoriser ou en interdire l'accès aux autres utilisateurs (voir les permissions sur les fichiers du cours précédent). Il existe un utilisateur spécial pour administrer un système Unix appelé : *administrateur*, *root* ou encore *super-utilisateur*. Contrairement aux utilisateurs classiques, ce compte à tous les droits sur le système. Il peut, entre autres, lire ou modifier tout fichier du système, ajouter/retirer des utilisateurs, installer de nouveaux logiciels ... On préfère donc utiliser ce compte utilisateur ponctuellement. Pour ce faire, on utilise la commande `sudo`.

Pensez à ne pas utiliser le même mot de passe pour le super utilisateur et les autres utilisateurs !

sudo *(Substitute User DO)* est une commande permettant à (presque) n'importe quel utilisateur d'exécuter des commandes qui ne peuvent être utilisée que par le super-utilisateur. Elle s'utilise comme suit :

`$ sudo` *`commande`*

puis tapez votre mot de passe utilisateur afin que le système vérifie que vous êtes bien qui vous prétendez être.

Sous Ubuntu le principe de gestion des super-utilisateurs est le suivant. Pour autoriser un utilisateur à devenir super-utilisateur, celui-ci doit se trouver dans le groupe `sudo` (voir le fichier `/etc/group`). Enfin, le fichier de configuration `/etc/sudoers` définit les autorisations et la configuration à laquelle on a accès quand on utilise la commande `sudo`.

On pourrait être tenté d'utiliser le super utilisateur au quotidien, afin de ne pas avoir à changer d'utilisateur lorsqu'il faut reconfigurer le système. Il ne faut pas faire cela !!! En effet, en tant que super utilisateur, une mauvaise manipulation peut causer la perte irrémédiable de tout ou partie de vos données ou rendre la machine inutilisable !

1.2 Utilisateur

Le fichier `/etc/passwd` contient toutes les informations relatives aux utilisateurs (login, interpréteur de commande, ...). C'est ce fichier que le système consulte lorsque vous vous connectez à votre compte en tapant votre identifiant et mot de passe. Si ce que vous avez tapé n'existe pas dans ce fichier alors vous ne pourrez pas vous connecter. Ce fichier respecte le format suivant :

```
nom_utilisateur : mot_de_passe : numero_utilisateur : numero_de_groupe : commentaire : dossier
: interpreteur_de_commandes
```

- `nom utilisateur` = identifiant de l'utilisateur
- `mot de passe` = mot de passe de l'utilisateur. Celui-ci est codé, il est inutile de l'éditer tel quel. Un bon mot de passe doit être long, dénué de sens, et doit contenir des caractères (majuscule et minuscule), chiffres et caractères spéciaux (voir cours n°1).
- `numero utilisateur` = un entier qui identifie l'utilisateur pour le système. Cet identifiant est unique. Les valeurs supérieures à 1000 sont pour les comptes utilisateurs. Il est appelé UID (User Identifier).
- `numero de groupe` = un entier qui identifie le groupe de l'utilisateur. C'est un identifiant unique appelé GID (Groupe Identifier).
- `commentaire` = des informations sur l'utilisateur pas utilisé par le système, mais utile pour l'administrateur).

- `dossier` = le dossier dans lequel se retrouve l'utilisateur après s'être connecté (dossier personnel).
- `interpreteur_de_commandes` = le programme exécuté quand vous lancez un terminal ou quand vous vous connectez en console.

Il pourrait ressembler à ceci :

```
root:x:0:0:root:/root:/bin/bash
nobody:x:99:99:Nobody:/:/sbin/nologin
tian:x:1000:1500:Tian:/home/tian:/bin/bash
tv:x:1001:1500:Television:/home/tv:/bin/bash
```

Pour interdire l'accès à un compte, il suffit de remplacer le mot de passe chiffré par une étoile : « * ». Les accès à un compte peuvent éventuellement être ouverts en laissant le champ ***mot_de_passe*** vide. Toute personne voulant se connecter avec ce compte pourra alors le faire.

Il est fortement déconseillé de laisser un compte sans mot de passe. En effet, si vous créez un tel compte, vous autorisés n'importe qui à se connecter sur votre ordinateur sans aucun mot de passe. Imaginez ce qui peut se produire si en plus cet utilisateur a accès à la commande sudo... Il pourra faire ce qu'il voudra sur votre machine !

Le super-utilisateur, lorsqu'il souhaite créer un nouveau compte utilisateur, peut utiliser soit l'interface graphique (menu **Paramètres Système** -> **Comptes utilisateurs**) de son système soit la commande :

```
$ useradd nom_utilisateur
```

Bien sûr, il faut également créer le dossier personnel de l'utilisateur. Pour ce faire, on utilise l'option `-m` de la commande `useradd`.

```
$ useradd -m nom_utilisateur
```

Cette dernière crée :

- le dossier personnel `/home/nom_utilisateur`
- la modification du fichier `/etc/passwd`

Pour attribuer un mot de passe à ce nouvel utilisateur, le super-utilisateur peut utiliser la commande suivante :

```
$ passwd nom_utilisateur
```

Pour supprimer un utilisateur, vous pouvez le faire à l'aide de la commande :

```
$ userdel nom_utilisateur
```

1.3 Devenir un autre utilisateur

1.3.1 En changeant de session

Pour se connecter sous le nom d'un utilisateur, vous pouvez vous déconnecter et vous reconnecter sous ce nouvel utilisateur (dans la barre de commande, sélectionner le profil et l'utilisateur souhaité dans la liste, ou sur le bouton démarrer, aller sur changer d'utilisateur pour ouvrir une nouvelle session sous cette nouvelle identité).

1.3.2 Dans la session courante : su

Si vous souhaitez juste changer temporairement d'identité dans un terminal, vous pouvez utiliser la commande su avec le nom d'utilisateur. Une fois validée, vous n'aurez plus qu'à taper le mot de passe de cet utilisateur pour vous identifier sous ce nom-là.

```
$ su user
```

1.3.3 Mais, au final, qui suis-je ? whoami

Il est important de savoir à tout moment quelle est l'identité sous laquelle nous allons exécuter une commande. Pour cela, nous avons la commande `whoami` qui nous renverra l'identité courante de l'utilisateur. L'utilisation de la commande `exit` vous permettra de quitter cette identité et de revenir à la vôtre (à la précédente en fait).

1.4 Groupes

Le fichier `/etc/group` contient la liste des utilisateurs appartenant aux différents groupes. En effet, lorsque de nombreux utilisateurs peuvent avoir accès au système, ceux-ci sont fréquemment rassemblés en différents groupes ayant chacun leurs propres droits d'accès aux fichiers et aux dossiers. Les entrées dans ce fichier **/etc/group** ont la syntaxe suivante :

```
nom_de_groupe : champ_special : numero_de_groupe : membre1, membre2
```

Le champ spécial est fréquemment vide. Le numéro de groupe est le numéro qui fait le lien entre les fichiers `/etc/group` et `/etc/passwd`.

Un même utilisateur peut apparaître dans plusieurs groupes. Lorsqu'il se connecte au système, il appartient au groupe spécifié dans le fichier `/etc/passwd` (le champ GID).

Pour ajouter un groupe, l'administrateur peut modifier le fichier */etc/group* à l'aide d'un éditeur de texte. Il peut également utiliser la commande :

```
$ groupadd nom_groupe
```

Dans le premier cas, il aura uniquement là où les lignes correspondant aux groupes, à ajouter. Pour ajouter un utilisateur à un groupe, il suffit d'éditer le fichier `/etc/group` et de rajouter ce nom au bout de la ligne en séparant le nom des membres par une virgule.

Pour supprimer un groupe, il suffit d'éditer le fichier `/etc/group` et d'effacer la ligne correspondante. Mais attention, il ne faut pas oublier de changer dans le fichier `/etc/passwd` les numéros (GID) du groupe supprimé.

Pour supprimer un groupe, vous pouvez le faire à l'aide de la commande :

```
$ groupdel nom_groupe
```

N. Ferry, S. Lavirotte, D. Lopez, H. Renard
Polytech'Nice – Sophia

TD séance n° 6
Utilisateurs et Groupes sous Unix et Windows

Exercices

Afin de réaliser ces exercices, vous allez devoir devenir super-utilisateur. Pour éviter de « casser » le système de la machine physique que vous utilisez, vous allez utiliser une machine virtuelle. Vous lancerez la commande `vm-linux` pour bénéficier de cette machine virtuelle.

Exercice n°1:

Allez vérifier le contenu du fichier `/etc/passwd`.

Exercice n°2:

- Votre compte d'utilisateur est-il défini dans le fichier `/etc/passwd` ?
- Quel est le dossier personnel de l'utilisateur `root` ?
- Quel est le programme interpréteur de commandes (Shell) de l'utilisateur `root` ?
- Quelle est la particularité de l'utilisateur `nobody` ?
- Quels sont les utilisateurs qui font partie du groupe `sudo` ? A votre avis quelle est la signification de ce groupe ?

Exercice n°3:

Créez un nouvel utilisateur `eve` avec la commande `useradd` (vous ne devez pas créer de home directory pour cet utilisateur). Quel(s) est (sont) le(s) fichier(s) modifié(s) lors de la création d'un nouvel utilisateur ?

Associez un mot de passe pour l'utilisateur `eve` à l'aide de la commande `passwd`. Quel(s) est (sont) le(s) fichier(s) qui sont modifiés (donnez la commande qui vous permet de le déterminer ? Quel est le fichier qui contient les mots de passe ?

Exercice n°4:

Créez l'utilisateur `walle` (en lui créant un dossier personnel) et associez-lui un mot de passe. Quels sont les fichiers qui ont été impactés dans `/etc` ? Y a-t-il eu des fichiers créés ailleurs, si oui, lesquels ? Visualiser le contenu du home-directory de l'utilisateur `walle` que l'on vient de créer.

Modifiez le fichier `/etc/passwd` pour que l'utilisateur `walle` utilise l'interpréteur de commande `/bin/bash`. Attention, il ne faut pas changer les permissions du fichier `/etc/passwd` !

Exercice n°5:

Créez un nouveau groupe `walle-movie`. Quel est l'impact de cette commande et sur quel(s) fichier(s) ?

Ajoutez les utilisateurs `walle` et `eve` au groupe `walle-movie`. Attention, il ne faut pas changer les permissions du fichier contenant la définition des groupes.

Exercice n°6:

Pour cet exercice, nous allons nous connecter sous le nom d'utilisateur `walle` que nous avons créé précédemment, sans fermer ou changer de session.

- Vérifiez sous quel nom d'utilisateur vous allez exécuter les commandes suivantes (cela doit être `walle`)
- Dans quel dossier vous trouvez-vous ? Pourquoi êtes-vous dans ce dossier-là ? Allez dans votre dossier d'utilisateur.
- Vous est-il possible de faire une copie du fichier `/etc/passwd` ? Vous est-il possible de supprimer ou de modifier le fichier `/etc/passwd` ? Expliquer la situation à l'aide de la commande `ls -l`.

TD séance n° 6
Utilisateurs et Groupes sous Unix et Windows

– Tentez de copier le fichier `/etc/shadow` dans votre dossier personnel. Cette opération est-elle possible ? Expliquez la situation à l'aide des permissions sur le fichier. A votre avis pourquoi peut-on copier le fichier `passwd` et pas le fichier `shadow` ?

Exercice n°7:

– Toujours en tant qu'utilisateur `walle`, dans votre dossier personnel, créez un fichier texte `file.txt`, qui soit lisible par tout le monde, mais non modifiable (même pas par vous). Essayer de lui ajouter un contenu à l'aide de l'éditeur de texte. Puis essayer de modifier le contenu du fichier en tant que super-utilisateur. Si vous rencontrez un problème, tentez de trouver comment le corriger.
– Créer un dossier nommé `secret`, dont le contenu est visible uniquement par vous-même.
– Les fichiers placés dans ce dossier sont-ils lisibles par d'autres membres de votre groupe ?

Exercice n°8:

Déconnectez-vous en tant qu'utilisateur `walle`, pour revenir à votre propre identité. Supprimez l'utilisateur `eve`. Supprimez l'utilisateur `walle` et les données de son compte personnel (son home directory) en une seule commande. Vérifiez quels sont les utilisateurs qui sont maintenant dans le groupe `walle-movie`. Supprimer le groupe `walle-movie`.

N. Ferry, S. Lavirotte, D. Lopez, H. Renard
Polytech'Nice – Sophia

TD séance n° 6
Utilisateurs et Groupes sous Unix et Windows

Exercices Complémentaires

Exercice A :

- Créez les utilisateurs `stage` et `toto`
- Effectuez des vérifications : possibilité immédiate de se loguer sous ces comptes, création de leur dossier personnel dans `/home`.
- Essayez de créer un compte déjà existant (recréer le compte `toto`)
- Supprimez sans regret les comptes de `stage` et `toto`. Son dossier personnel a-t-il été supprimé ?

Exercice B :

Il s'agit de créer un groupe nommé `stagiaire` dont les membres sont les comptes `stageX`. On donnera ensuite à ce groupe des droits complets sur un dossier partagé.

- Créez le groupe `stagiaire`
- Ajoutez quelques comptes `stageX` dans ce groupe (stage 1, stage 2 et stage3)
- Vérifiez le résultat avec la commande `groups`
- Créez un dossier `partage` dans lequel seul les gens du groupe stagiaire peuvent lire et écrire
- Vérifiez que seul les comptes `stageX` peuvent bien créer un fichier dans
- Supprimez le groupe stagiaire et les utilisateurs stage1, stage2 et stage3