TP n°1: Gestion des utilisateurs

Objectifs
A l'issue de ce chapitre, vous serez en mesure de gérer les comptes des utilisateurs et leurs groupes. Vous connaitrez également la structure des fichiers qui contiennent les informations fondamentales.

Contenu
Les fichiers */etc/passwd* et */etc/group*
Les commandes d'administration: useradd, passwd, su, id,

1- Rappels de cours: Structure des fichiers /etc/passwd et /etc/group

1.1- Exemples:

$ **more** /etc/passwd
rootCuSzE74h021 CS :0:0: Super User:/:/bin/bash
ali: :200:50:Ali Ali:/horne/ali:/bin/csh
brahim: OZkhUrn Yrprtpt20 1:50:Brahim Brahim :/horne/brahim:/bin/sh

$ **more** /etc/group
root::0:root
group::50:ali,brahim

1.2- Structure du fichier */etc/passwd*
Le fichier */etc/passwd* est un fichier de type texte dont chaque ligne définit un compte utilisateur. La ligne est composée de champs. Le séparateur de champs est le symbole « : ».

Nom de connexion	:	Mot de Passe	:	UID	:	GID	:	Commentaire	:	Répertoire de connexion	:	Commande de connexion

Nom de connexion: saisi lors de la demande de connexion

Mot de passe: présent dans le fichier, mais crypté. Dans les systèmes sécurisés, le champ mot de passe existe toujours mais il contient le caractère x. Le mot de passe crypté est déporté dans un fichier accessible au seul administrateur. Les caractères "!!" dans le champs mot de passe indiquent que le compte n'est pas accessible.

UID: L'administrateur attribue un numéro à chaque utilisateur. Ce numéro, l'UID (« *User Identification* »), est mémorisé dans les descripteurs de fichiers pour en identifier le propriétaire. C'est donc l'information pertinente, utilisée par le système Linux, pour identifier un utilisateur. L'UID de *root* est 0.

GID: L'administrateur identifie le groupe de connexion d'un utilisateur grâce au champ GID (« *Group Identification* »). Le fichier */etc/group* associe un nom de groupe à ce GID et définit les groupes supplémentaires de l'utilisateur.

Commentaire: La zone est utilisée librement par l'administrateur pour commenter le compte. Il peut être structuré en suivant les recommandations de la commande finger. On y trouve, entre autres, le nom et le prénom.

Répertoire de connexion: Ce champ détermine le répertoire de connexion de l'utilisateur, conventionnellement */home/ali* pour l'utilisateur de nom de connexion ali. Ce répertoire contient les fichiers de configuration (*.bash_profile)* de l'utilisateur.

Commande de connexion: Ce champ précise le chemin d'accès absolu de la commande à exécuter lors de la connexion. C'est généralement un shell.

Remarques:
• L'UID est une valeur comprise entre 0 et la valeur définie par la constante UID_MAX du fichier /etc/login.defs. Les valeurs inférieures à 100 sont généralement réservées pour des utilisateurs associés à des services standard du système Linux. La constante UID_MIN du fichier /etc/login. defs définit la valeur minimale des UID des utilisateurs.

• L'attribution d'un UID est de la responsabilité de l'administrateur et rien ne l'oblige à les affecter séquentiellement. Il peut définir sa propre stratégie.

• Quand un administrateur gère un parc de machines Linux sans mettre en oeuvre d'administration centralisée, il est conseillé d'attribuer le même UID à un utilisateur qui possède un compte sur plusieurs machines du réseau.

• Si plusieurs lignes utilisent le même UID pour plusieurs noms de connexion différents, un seul utilisateur est en fait défini. On peut ainsi définir un utilisateur stop, dont l'UID est 0 et qui exécute shutdown comme commande de connexion.

• Dans un système non sécurisé, le mot de passe peut être absent, ce qui permet de se connecter sans avoir à fournir de mot de passe.

1.3- Structure du fichier */etc/group*

Le fichier /etc/group est un fichier de type texte dont chaque ligne définit un groupe d'utilisateurs, La ligne est composée de champs. Le séparateur de champs est le symbole « : ».

Nom du groupe	:	Mot de passe	:	GID	:	Liste des utilisateurs autorisés à se connecter au Groupe Syntaxe: (util [,util ...])

Nom du groupe: Le nom du groupe est celui utilisé dans la commande newgrp ou affiché par la commande ls.

Mot de passe: Le mot de passe est présent dans le fichier, mais crypté. Il est demandé à un utilisateur qui veut se connecter au groupe et qui ne figure pas dans la liste des utilisateurs du groupe.

Liste des utilisateurs: La liste des utilisateurs qui peuvent se connecter au groupe par la commande newgrp sans avoir à fournir de mot de passe.

La commande newgrp permet de changer le groupe de référence utilisé lors de la création de nouveaux fichiers. A défaut d'avoir exécuté la commande newgrp, c'est le groupe de connexion qui est utilisé.

Remarques:

• Un utilisateur n'a pas besoin d'être mentionné dans la liste des utilisateurs de son groupe de connexion.

• Le champ mot de passe est rarement utilisé dans la pratique .

2- Activités de préparation au TP

Les commandes de gestion des utilisateurs

• useradd, usermod, userdel :	Gèrent les comptes utilisateur
• groupadd, groupmod, groupdel :	Gèrent les comptes de groupe
• pwck, grpck :	Vérifient les fichiers */etc/passwd* et */etc/group*
• finger:	Donne des informations sur un utilisateur
• chfn, chsh :	Changent le shell ou le commentaire d'un utilisateur
• passwd :	Permet de modifier le mot de passe d'un utilisateur
• su :	Permet de se connecter à un compte
• id :	Permet de connaître son identité
• groups :	Donne la liste des groupes d'un utilisateur
• vipw, vigr :	Edite les fichiers */etc/passwd* et */etc/group,* en les verrouillant

Les commandes de gestion des comptes d'utilisateurs et de groupes sont nombreuses.
Les commandes d'administration proprement dites: useradd, usermod, userdel, groupadd, groupmod, groupdel, pwck et grpck peuvent être utiles quand elles sont intégrées à un script. Elles ne présentent pas de difficultés d'emploi. Dans la réalité, il est souvent bien plus pratique de procéder aux opérations grâce à l'outil intégré d'administration linuxconf qui propose, pour chaque paramètre d'un compte, le choix par défaut le mieux adapté. L'outil effectue aussi une vérification des éléments fournis. Attention, quelques-unes des options de la commande useradd, réservée à l'administrateur, permettent de paramétrer le compte. La commande useradd permet en effet, en sus de la création de comptes d'utilisateurs, de définir les paramètres qui seront utilisés, par défaut, à la création d'un compte.

2.1- La commande useradd:

- *Syntaxe pour la création d'un compte:*

useradd [-e comment] [-d home_dir] [-e expire_date] [-finactive_time] [-g initial group]
[-G group] [, ...]] [-m [-k skeleton_dir]] [-s shell] [-u uid [-o]] [-n] [-r] login

Les options sont celles dont la seule lecture ne donne pas l'explication:

-c comment:	Le commentaire.
-d home_dir:	Le répertoire de connexion.
-e expire_date:	La date d'expiration du compte.
-f inactive_time:	Le nombre de jours au bout duquel un compte est inutilisable, après l'expiration d'un mot de passe.
-g initial group:	Le groupe initial, par défaut, ali pour l'utilisateur ali.
-G group,.. :	Les groupes supplémentaires.
-m:	Il faut créer le répertoire de connexion de l'utilisateur.
-k skeleton _dir:	Le répertoire de peuplement du répertoire de connexion. Les fichiers qu'il contient sont recopiés dans le répertoire de connexion de l'utilisateur. C'est /etc/skel par défaut.
-s shell:	Le shell de l'utilisateur, par défaut bash.
-u uid:	L'UID de l'utilisateur.
-n:	Un groupe du nom de l'utilisateur n'est pas créé.
-r:	La commande accepte de créer un compte avec un UID inférieur à UID_MIN, défini dans /etc/login.defs.

Remarques:
• Le répertoire /etc/skel de Linux est très important: il contient des modèles de fichiers de configuration des sessions des utilisateurs.

• L'administrateur peut ajouter des fichiers ou les modifier pour définir les paramètres communs à tous les utilisateurs du site, comme les options du shell ou la définition des touches d'édition du shell bash. Ces fichiers sont automatiquement copiés dans le répertoire de connexion des utilisateurs créés par la commande useradd ou l'outil d'administration linuxconf.

Exemple:

```
# ls -aC /etc/skel
.Xdefaults .bash profile .inputrc
.bash_logout .bashrc
# useradd ali
# ls -aC ~ali
.Xdefaults .bash -profile .inputrc
.bash_logout .bashrc
```

- *Syntaxe pour la définition des paramètres par défaut*

```
useradd -D [-g default group] [-b default_home] [-fdefault_inactive]
                [-e default_expire_date] [-s default_shell]
```

Dans cette forme, la commande useradd permet de définir les valeurs utilisées par défaut quand on crée un compte utilisateur.
La commande useradd -D visualise les valeurs actuellement utilisées.

2.2- Les autres commandes

Les commandes d'information ou de gestion courantes, utiles en mode commande: finger, users, groups, id (voir les Exemples) et su.

Les commandes générales: vipw, vigr, passwd.

La commande vipw réalise l'édition du fichier /etc/passwd. Elle en effectue d'abord le verrouillage pour en garantir un usage exclusif. L'éditeur de texte qui est exécuté est défini par la variable d'environnement EDITOR, vi à défaut. Si le verrouillage du fichier n'est pas possible, la commande vipw le signale et demande à l'administrateur d'essayer plus tard. Le fichier est automatiquement déverrouillé à la fin de l'édition. La commande vigr agit de même avec le fichier /etc/group.

2.3- La commande passwd

La gestion des mots de passe et de leur pérennité (« *Aging Information* ») est réalisée par la commande passwd, déjà connue des utilisateurs, et par la commande chage. Le rôle principal de la commande passwd est de créer ou de modifier le mot de passe d'un utilisateur. La commande chage gère la pérennité des mots de passe.

La commande passwd a plusieurs fonctions pour l'administrateur:
• Modifier le mot de passe d'un utilisateur:

```
# passwd Nom_utilisateur
```

• Supprimer le mot de passe d'un utilisateur:

```
# passwd -d Nom _utilisateur
```

• Verrouiller le compte d'un utilisateur, ce qui empêche sa connexion:

```
# passwd -l Nom_utilisateur
```

• Déverrouiller le compte d'un utilisateur:

```
passwd -u Nom_utilisateur    # ou passwd -d Nom_utilisateur
                             # ou passwd Nom_utilisateur
```

Remarques:
• Les restrictions imposées aux utilisateurs dans la définition de leur mot de passe ne s'appliquent pas à l'administrateur qui peut attribuer n'importe quel mot de passe à un utilisateur, sauf une chaîne vide .
• En l'absence de mot de passe, l'invite « password » n'est pas affichée à la connexion.

Exemples:
# id
uid=0(root), gid=0(root)
groups=0(root), 1(bin), 2(daemon), 3(sys), 4(adm), 6(disk), 10(wheel)
# finger

Login	Name	Tty	Idle	Login	Time	Office	Office Phone
ali	ali	-	Br 4	Apr 21	17:01		
root	root	*2	3	Apr 21 13:30			

users
ali root
groups
root bin daemon sys adm disk wheel
groups ali
ali: ali compta

Attribuer à "ali" le mot de passe ali
New UNIX password:
BAD PASSWORD: it is too short

Supprimer le mot de passe de ali
passwd -d ali

Verrouiller le compte de ali
passwd -l ali
ali ne peut plus se connecter
venus login : ali
Password:
Login incorrect
venus login :

Déverrouiller le compte de ali
passwd ali
Changing password for user ali
New UNIX password:
Retype new UNIX password:
passwd: ail authentication tokens updated successfully
ali peut à nouveau se connecter
venus login : ali
Password:
/home/ali $

Visualiser les attributs par défaut
# useradd -D
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=Ibin/bash

SKEL=/etc/skel

Ajouter un utilisateur avec les paramètres par défaut de /etc/login.defs
useradd brahim

Attribuer le mot de passe unix98 à brahim
passwd brahim
New UNIX password:
BAD PASSWORD: it is based on a dictionary word
Retype new UNIX password:
passwd: ail authentication tokens updated successfully
grep brahim /etc/passwd
brahim:x:505 :505::/home/brahim:/bin/bash

Connexion de brahim
Fedora Core Linux release 9.2
Kemel 2.6.7-10 on an i686venus
login:brahim
password:
/home/brahim $

Créer les groupes develop, compta et achats, de GID: 1000, 1001 et 1002
groupadd -g 1000 develop
groupadd -g 1001 compta
groupadd -g 1002 achats

Créer un utilisateur avec des paramètres spécifiques
useradd -u 2000 -s /bin/bash -d /home/siham -c "Siham - Casa" siham

Visualiser le compte créé précédemment
grep siham /etc/passwd
siham:!! :2000:2000:Siham - Casa:/home/siham:/bin/bash
grep siham /etc/group
siham:x:2000:

Ajouter siham aux groupes compta et develop
usermod -G compta,develop siham

Lister les groupes de siham
groups siham
siham: siham develop compta

Supprimer l'utilisateur siham (l'option -r demande la suppression de son arborescence)
userdel -r siham

3- Enoncé du TP (à réaliser par les étudiants sous la supervision du professeur)

TP n°1: Gestion des utilisateurs

Objectifs:
• Savoir créer un compte utilisateur
• Savoir gérer les utilisateurs et les groupes

Exercice 1:
Est-ce que l'utilisateur bin existe, si oui, quel est son uid ?
Exercice 2:
Comment feriez-vous pour vous connecter sous le compte de l'utilisateur « bin » ?
Exercice 3:
Existe-t-il d'autres comptes utilisateurs possédant les droits de *root* ?
Exercice 4:
A quels groupes appartient l'utilisateur bin ?
Exercice 5:
Créez avec useradd, en gardant toutes les valeurs par défaut, l'utilisateur "ali". Quel est le groupe de "ali"?
Exercice 6:
Ajoutez "ali" au groupe staff. Au besoin, créez ce groupe.
Exercice 7:
Affichez les groupes de l'utilisateur "ali".
Exercice 8:
Connectez-vous au compte "ali" nouvellement créé de deux manières, à la connexion et grâce à la commande su. Expliquez les deux résultats.
Exercice 9:
Que faut-il faire pour pouvoir se connecter au compte "ali"?
Exercice 10:
Changez le champ commentaire de "ali" en utilisant la commande vipw. Renseignez le champ avec le texte suivant: "Ali BenBrahim - Rabat".
Exercice 11:
Créez un compte utilisateur avec l'outil d'administration (linuxconf, ...).
Exercice 12:
Créez un compte « admin » d'UID 0. Il peut servir si, par exemple, on oublie le mot de passe de root