**La gestion des utilisateurs et des groupes**

**Les différents types de comptes**

Les comptes utilisateur ne sont pas tous égaux sur Linux. On distingue trois types:

**Superutilisateur**

L'utilisateur **root** est le plus important du système d'exploitation Linux.

Il n’est pas concerné par les droits d’accès aux fichiers.

Ce super-utilisateur aura donc à sa charge les tâches d’administration du système.

Le UID de l'utilisateur root est égal à 0 (zéro).

**Comptes systèmes (bin, daemon, sync, apache…)**

Plusieurs comptes ne sont pas affectés à des utilisateurs, ils servent à faciliter la gestion des droits d’accès de certaines applications et démons.

Par exemple, le serveur Web utilise l’identité du compte "apache".

**On ne doit jamais utiliser un serveur Web sous l’identité de root.**

Pour des raisons de sécurité, on fera en sorte que personne ne puisse se connecter à l'ordinateur à partir de l’un de ces comptes.

Les UID compris entre 1 et 499 sont utilisés pour les comptes systèmes.

**Comptes ordinaires**

Tous les autres comptes utilisateur sont associés à des personnes; leur vocation est de permettre à des utilisateurs de type standard de se connecter.

Le UID d’un utilisateur sera un nombre supérieur ou égal à 500.

**La commande su permet de se connecter à un autre compte**

*Note importante sur « su » : Ubuntu version Destktop a verrouillé le compte root*

ne donne pas accès aux variables d'environnement du root

* **su**

donne accès aux variables d'environnement du root

* **su -**

ne donne pas accès aux variables d'environnement de u1

* **su u1**

donne accès aux variables d'environnement de u1

* **su - u1**

permet d'exécuter une commande en tant que root

* **su -c ifconfig**
* **su -c "ls -l /bin/"**

pour éviter d'écrire la commande sudo à chaque fois

**su -i** L'utilisation de cette commande n'est pas recommandée à moins d'être un expert.

**Création d'un compte utilisateur**

La commande **adduser** permet d’ajouter un nouvel utilisateur de façon plus interactive. Cependant, useradd permet plus facilement de manipuler la création des usagers.

**création d'un utilisateur sans spécifier son UID**

* # useradd u1 -m -s /bin/bash

ajoute une ligne à la fin du fichier /etc/passwd

un UID est assigné à l'utilisateur u1

par défaut le groupe principal de l'utilisateur porte le nom de l'utilisateur

crée un répertoire personnel pour l'utilisateur /home/u1

copie le contenu du répertoire **/etc/skel/** dans **/home/u1/**

-m = permet de créer son home

-s = permet de spécifier son shell (son terminal par défaut)

UID

User : id nom\_du\_user

Groupe : id -g nom\_du\_groupe

**Modification d'un compte utilisateur**

La commande **usermod** permet de modifier les paramètres d'un compte utilisateur.

Pour verrouiller le compte d'un utilisateur

* usermod -L et1

En considérant que le groupe "spy" existe, quelle est la commande pour l'ajouter comme groupe secondaire à l'usager king?

usermod -aG spy king

adduser king spy

sudo usermod -a -G spy king

**Destruction d'un compte utilisateur**

La commande **deluser** permet d'effacer un compte utilisateur.

**Les différents types de groupes**

Il existe différents types de groupes permettant de donner des droits communs à un ensemble d’utilisateurs.

**Groupe root**

C’est le groupe principal de l’administrateur (**root**).et son GID est 0.

Le GID du groupe root est égal à 0 (zéro).

**Groupes systèmes (bin, daemon, sync, apache**…)

Ces groupes jouent le même rôle que les comptes du même nom et permettent de donner les mêmes droits d’accès à un ensemble d’applications.

Les groupes système auront un GID compris entre 1 et 499.

**Groupes ordinaires**

Ces groupes représentent un ensemble de personnes réelles devant accéder aux mêmes fichiers.

Les groupes ordinaires auront un GID supérieur ou égal à 500.

**Groupe principal**

Un utilisateur est obligatoirement membre d'un groupe d'utilisateurs sur un système LINUX, son groupe principal sera utilisé lors de la création des fichiers.

**Groupe secondaire**

Un utilisateur peut éventuellement appartenir à plusieurs autres groupes: ses groupes secondaires détermineront ses droits d'accès aux fichiers créés par d'autres membres de ces groupes.

**Création d'un groupe**

La commande **groupadd** permet d'ajouter un groupe.

**groupadd (pour ajouter un groupe)**

* # groupadd g33

ajoute une ligne à la fin du fichier /etc/group

**Modification d'un groupe**

La commande **groupmod** permet de modifier les paramètres d'un groupe.

**Destruction d'un groupe**

La commande **groupdel** permet d'effacer un groupe.

**La commande groups**

La commande **groups** affiche l'appartenance des groupes d'un utilisateur

* groups
* groups u1

Quelle est la commande pour voir tous les groupes de l'usager king?

Groups king

**La commande id**

La commande **id** affiche des informations sur le compte d'un utilisateur.

Le id d`un group grep '^nom\_du\_groupe:' /etc/group

id

id u1

id -g

id -gn

id u1 -g

id u1 -gn

1. La commande **id** sans argument affiche les informations d'identification de l'utilisateur courant, y compris son identifiant utilisateur (UID), ses groupes d'appartenance et d'autres informations.
2. La commande **id u1** affiche les informations d'identification de l'utilisateur "u1", y compris son UID, ses groupes d'appartenance et d'autres informations.
3. La commande **id -g** affiche uniquement l'identifiant de groupe principal de l'utilisateur courant.
4. La commande **id -gn** affiche uniquement le nom du groupe principal de l'utilisateur courant.
5. La commande **id u1 -g** affiche uniquement l'identifiant de groupe principal de l'utilisateur "u1".
6. La commande **id u1 -gn** affiche uniquement le nom du groupe principal de l'utilisateur "u1".

id -G affiche tous les identifiants de groupe auxquels l'utilisateur "kingsman" est affilié.

id -Gn affiche tous les noms de groupe auxquels l'utilisateur courant est affilié.

id u1 -G affiche tous les identifiants de groupe auxquels l'utilisateur "u1" est affilié.

id u1 -Gn affiche tous les noms de groupe auxquels l'utilisateur "u1" est affilié.

La commande id -Gn User affiche tous les noms de groupe auxquels l'utilisateur "User" est affilié.

Lorsque vous exécutez cette commande, la sortie affiche une liste séparée par des espaces des noms de groupe auxquels l'utilisateur "User" appartient.

La commande id u1 -GUser est incorrecte et n'est pas une syntaxe valide pour la commande id. Pour afficher les identifiants de groupe auxquels l'utilisateur "u1" est affilié, vous pouvez utiliser la commande id u1 -G.

Lorsque vous exécutez cette commande, la sortie affiche une liste séparée par des espaces des identifiants de groupe auxquels l'utilisateur "u1" est associé.

La commande id u1 -Gn affiche tous les noms de groupe auxquels l'utilisateur "u1" est affilié.

Lorsque vous exécutez cette commande, la sortie affiche une liste séparée par des espaces des noms de groupe auxquels l'utilisateur "u1" appartient.

En résumé, la différence entre les trois commandes réside dans les arguments fournis à la commande id. La première commande spécifie un nom d'utilisateur pour afficher les noms de groupe, la deuxième commande contient une syntaxe incorrecte, et la troisième commande spécifie un nom d'utilisateur pour afficher les noms de groupe.

**La commande newgrp**

La commande **newgrp** permet de changer de groupe principal pour la session.

**Les fichiers importants pour la gestion des usagers et des groupes sont:**

* /etc/passwd contient la liste des comptes utilisateur
* /etc/group contient la liste des groupes
* /etc/shadow contient la liste des mots de passe cryptés

Cat /etc/shadow

Commande pour obtenir des informations sur la structure du fichier /etc/passwd

* man 5 passwd

Commande pour obtenir des informations sur la structure du fichier /etc/shadow

* man 5 shadow

Commande pour obtenir des informations sur la structure du fichier /etc/group

* man 5 group

**Le champ numéro 2 du fichier /etc/shadow contient le "mot de passe"**

**:$1$ff5t6&%:** 🡺 le premier caractère est $ donc le champ contient un mot de passe

**:!!$1$ff5t6&%:** 🡺 le compte est verrouillé par la commande **passwd**

**:!$1$f5t6&%:** 🡺 le compte est verrouillé par la commande **usermod**

**:!!:** 🡺 le mot de passe n'a pas encore été défini

**:\*:** 🡺 impossible de se connecter sur la station avec ce compte

**::** 🡺 aucun mot de passe ne sera demandé lors du login

**note 1: Le mot de passe peut commencer par un caractère "$". Ce qui veut dire que le mot de passe a été généré par un autre algorithme que DES.**

**note 2: Si le mot de passe débute par $1$ l’algorithme basé sur MD5 a été utilisé.**

**La commande passwd**

Pour attribuer un mot de passe à l'utilisateur et1

* passwd et1

Pour enlever le mot de passe d’un utilisateur

* passwd -d et1

Pour attribuer automatiquement un mot de passe à un utilisateur

* echo -e "123\n123" | passwd et1

Pour verrouiller le compte d'un utilisateur

* passwd -l et1

Pour déverrouiller le compte d'un utilisateur

* passwd -u et1

**La commande chage**

Cette commande modifie les informations de validité d’un mot de passe.

Permet de forcer l'utilisateur et1 à changer son mot de passe au prochain login.

* chage -d 0 et1

Affiche les informations sur les âges du compte et1.

* chage -l et1

**newusers**

En premier on crée la liste des usagers dans un fichier "**liste.usagers**".

usager1:secret1:525:525::/home/usager1:/bin/bash

usager2:secret1:526:526::/home/usager2:/bin/bash

...

note: la commande newusers va crypter le mot de passe

**newusers liste.usagers**

**chpasswd**

En premier on crée la liste des mots de passe dans un fichier "**liste.motdepasse**"**.**

Le fichier "**liste.motdepasse**" doit contenir le nom de l'utilisateur et son mot de passe en respectant la syntaxe suivante "**username:password**", une paire par ligne.

usager1:passwd1

usager2:passwd2

...

**cat liste.motdepasse | chpasswd**