USERS AND GROUPS MANAGEMENT

1.- Basic ideas about system resources, users, user accounts and groups

1.1.- System resources, users and users accounts

a) A system resource is a component or element within a computer system that is utilized to accomplish a goal such as reading a document, storing data, downloading a file , watching a video clip, playing a game or performing a mathematical operation.

A system Resources can be:

- Hardware components such as CPU, RAM memory, network interfaces and storage devices
- Software components such as applications, services and operating systems.
- Files such as databases, pictures, text documents and so on.
- Folders

Like any resource, computer system resources can be exhausted, and issues arise due to scarcity.

b) In Linux, a **user** is an individual or system entity (such as a program in execution or a hardware device) that can access to the system and its resources. Because users can access to resources, and resources are always scarce, managing users and their access permissions to resources is a key aspect of administering a computer system.

c) Each user has a user account. A user account includes the information a user needs to use the system and its resources. A user account can represent a human being or a system entity (such as a program in execution or a hardware device).

d) There are several types of user accounts in computing systems:

- System accounts are created by the operating system and are used to run programs (such as system services and standard processes). These accounts have elevated access privileges to access system resources but are not used for interactive login.
- Administrator accounts have full access permissions to make changes to the system. They are used to install software, configure settings, add or remove user accounts, and perform other administrative tasks. Administrator accounts should be limited to authorized personnel only.
- Standard user accounts accounts have basic access permissions to normal system resources and are used by general system users to login and perform routine tasks. They have limited permissions to make system changes.
- Local accounts are stored on the local system and provide access only to that system.
- Network accounts are stored on a network domain controller and provide access to resources on the network.
- Remote accounts allow users to login to a system from a remote location over a network. Extra security measures should be implemented for remote access to safeguard systems and data.

d) Proper configuration and management of user accounts are crucial for:

- Maintaining system security and preventing unauthorized access to resources.
- Managing efficiently the system resources.
- Ensuring system estability.

d) Typical components of a user account are: user (or login) name, password, user identification number, group membership, home directories, expiration dates and other information.

e) On Linux, user account information is stored in **/etc/passwd** and **/etc/shadow** files.

1.2.- Groups

a) Groups are just lists of user accounts.

b) Users can be organized into groups, which helps simplify permission management. Groups are used as a basis for determining file access permissions. Group membership allows some users to access resources such as folders, files, and hardware that others users cannot access.

d) Typical components of a group are: group name, group identification number and a list of users members of the group.

e) On Linux, group information is stored in **/etc/group** and **/etc/gshadow** files.

2- Commands for managing users accounts and groups

2.1.- mkpasswd

a) Descrition: The command-line utility **mkpasswd** encrypts a given password. This command is part of a package called **whoi**s. You have to install **whois** in order to install **mkpasswd** on your system.

b) Synopis: **mkpasswd PASSWORD**

c) As a result, a encrypted version of PASSWORD will be displayed on screen

d) Example: If you want to create an encrypted version of **FjeClot@25** run → **mkpasswd FjeClot@25**

2.2.- useradd

a) Descrption: The **useradd** command adds a new user to the system. This command adds a new entry to **/etc/passwd** and **/etc/shadow**.

b) Synopsis: **useradd username <options>**

c) Important options:
-u --> The User IDentifier or UID. By default, will be the first free ID after the greatest used one.
-g --> The group name or number of the user's default group (or primary group). The group name or number must refer to an already existing group. If not specified, the default from /etc/default/useradd is used.
-d --> It specifies the user personal directory. If not specified, the default from /etc/default/useradd is used.
-m --> If it does not exist, the home directory for the new user account will be created.
-s --> Specify user's login shell. The default for normal user accounts is taken from /etc/default/useradd.
-k --> The skeleton directory, by default /etc/skel, that contains files and directories to be copied in the user's home directory when the home directory is created by useradd. This option is only valid if the -m option is specified.
-G --> A list of supplementary groups which the user is also a member of.
-p --> Encrypted password as returned by mkpasswd.

d) Example:

If you want to add to the system a new user called **tux**, with the following characteristics: **uid=1001**, default group=**users**, personal directory=**/home/tux**, default shell=**/bin/bash**, skeleton directory=**/etc/ske**l, additional groups=**adm,sys** and password = **Clot8Fje@**9 , you should run the following single command:

useradd tux -u 1001 -g users -d /home/tux -m -s /bin/bash -k /etc/skel -G adm,sys -p $(mkpasswd Clot8Fje@9)

2.3.- userdel

a) Description: The **userdel** command deletes an user account.

b) Synopsis 1: **userdel username** ==> The user will be deleted but not its home folder. Entries in /etc/passwd, /etc/shadow and /etc/group will be deleted. Folder /home/username will not be deleted.

c) Synopsis 2: **userdel -r username** ==> The user will be deleted and its home folder as well. Entries in /etc/passwd, /etc/shadow and /etc/group will be deleted. Folder /home/username will be deleted.

d) Example: If you want to completely remove the user **tux**, run the following command → **userdel -r tux**

2.4.- usermod

a) Description: The **usermod** command modifies an user account.

b) Synopsis: **usermod <options> username**

c) Important options:
-d --> This option specifies the new home directory of the user.
-g --> The group name or number of the user's new default group.
-p --> An Encrypted new password.
-s --> Specifies an user's new login shell.
-u --> Changes the User IDentifier or UID.
-a -G --> Adds a user to one or more groups

d) Example 1:

The following command changes the UID. The new UID will be 590:

```
usermod -u 590 tux
```

e) Example 2:

The following command changes the password. The new password will be ClotFJE@91:

```
usermod -p $(mkpasswd ClotFJE@91) tux
```

f) Example 3:

The following command changes the UID and password. The new password will be ClotFJE@91 and the new UID will be 620:

```
usermod -p $(master2013) -u 620 tux
```

2.5.- groupadd

a) Description: The **groupadd** command adds a new group. This command adds a new entry to **/etc/groups** and **/etc/gshadows**.

b) Synopsis: `groupadd <options> group_name`

c) Important options:
-g --> The Group IDentifier or GID. By default, will be the first free ID after the greatest used one.

d) Example:

The following command adds a new group called students. The value assigned to GID will be 120:

```
groupadd -g 120 students
```

2.6.- groupdel

a) Description: The **groupdel** command deletes a group.

b) Synopsis: `groupdel group_name`

c) Example: If you want to remove a group called students run → `groupdel students`

d) Important: A user's default group (also called primary group) is not removeable. Delete the user account or modify its primary group if you want to delete that group.

2.7.- groupmod

a) Description: The **groupmod** command modifies a group using the values specified on the command line.This command can modifiy entries in **/etc/groups** and **/etc/shadow**.

b) Synopsis: `groupmod <options> group_name`

c) Important options:
-g → Changes the Group IDentifier or GID
-n → Changes the group name

d) Example 1:

The following command changes the GID. The new GID will be 190

```
groupmod -g 190 students
```

e) Example 2

The following command changes a group name from the old name is students to new one called teachers:

```
groupmod -n teachers students
```

2.8.- gpasswd

a) If you want to:

- Remove a user from a group, run the command: **gpasswd -d username group_name**
- Add a user to group, run the command: **gpasswd -a username group_name**

b) Example 1:

The following command adds a user to a group. A new user called teacher02 will be added to a group called teachers if you run:

gpasswd -a teacher02 teachers

c) Example 2:

The following command deletes a user from a group. A user called teacher02 will be deleted from a group called teachers if you run:

gpasswd -d teacher02 teachers

2.9.- members

a) Description: The **members** command outputs a list of users members of a group.

b) Important: The **members** command is not installed by default. Install **members** with the help of **aptitude**.

c) Synopsis: **members -a group_name**

d) Example: The following command show a list of users members of group **sudo** -> **members -a sudo**

2.10.- id

a) Description: The **id** command shows user identifier, user name, primary group identifier, primary group name and all the groups identifiers and names the user belongs to.

b) Synopsis: **id user_name**

c) Example → **id dacomo**

3- User and group files

a) The **/etc/passwd** file contains information about all system users. A typical line in **/etc/passwd** looks like :

dacomo:x:1000:100::/home/dai1:/bin/bash

Where the system stores a **username**, a **password** in plain text or a x if it is encrypted and saved in **/etc/shadow** special file, the user **uid**, the **gid** of the user primary default group, the user **home directory** and the **shell** that runs when user connects.

b) The **/etc/group** file contains information about all system groups. A typical line in **/etc/group** looks like :

vboxusers:x:122:dacomo,asix,daw

Where the system stores a **group name**, a **password** in plain text or a x if it is encrypted and saved in **/etc/gshadow** special file, the group **gid**, and a **list** of group **members**.

c) For each line of /**etc/shadow**, there is a **username** and the **encrypted version of the user's password** (and other items that do not concern us now). If the password field contains **!** or *, the user will not be able to use a password to log in (but the user may log in the system by other means).

d) For each line of /**etc/gshadow**, there is a **group name** and the **encrypted version of the group's password**. Additionally, a list of group members is stored for each group. If the password field contains ! or *, users will not be able to use a password to access the group. The group's password is rarely (or never) used nowadays.