TP n°7: User and group management

1. Objectives

The aim of this practical course is to familiarize students with the simple commands for managing users and groups.

2. The various files containing information about users or groups

A user's account is represented by a login and an associated password. Information about the user accounts available on a Unix machine is grouped together in the file "***/etc/passwd***". Each line in this file corresponds to an account and is made up of 7 fields separated by :.

2.1 The file "/etc/passwd"

The file "***/etc/passwd***" contains information about all the users on the system. Each line in the file relates to one user. The various fields are separated by a colon (:) as follows:

```
login:password:UID:GID:user-info:home-directory:shell
```

- *login*: is the identifier that the user must enter to authenticate.
- *password*: is the password encrypted in md5 (software that generates a hash, i.e. an image for the password, and is used to check the integrity of the password). This field can have several meanings:
  - *: it is impossible to authenticate on the system with this account.
  - !! the account is deactivated.
  - x or ! the password is in a shadow file.
  - empty field: there is no password for this account.
- *UID*: unique identifier for the user, in the form of a numerical value. The root user has UID 0 and UIDs below 100 are reserved for system accounts.
- *GID*: identifier for the user's main group.
- *user-info*: full name and other information such as extension number... Each piece of information is separated by a comma.
- *home-directory*: directory in which the files belonging to the user are stored. Usually in the form /home/toto.
- *shell*: indicates which shell will be launched after authentication.

The file *"/etc/passwd"* is accessible to all users because some commands need to know the list of users or the login/UID correspondence.

2.2 The files "/etc/shadow" and "/etc/gshadow"

Passwords are stored in the file *"/etc/shadow"* for users and in the file "/etc/gshadow" for groups. These two files can only be read by the administrator. With the file "shadow", Linux offers the possibility of managing a password ageing system where the administrator can set a password time limit after which the user is invited to change their password or risk losing access to their account.

A line in the file *"/etc/shadow*" consists of nine fields separated by the (:) character.

- Login,
- Password: An * in this field indicates that the account cannot be connected (as in the case of the account “bin”, for example). A password beginning with !! indicates that the account is locked.
- Age: The number of days since 1 January 1970 and the date the password was updated.
- Change period: The minimum number of days between two password changes. A 0 indicates that the user can change the password at any time.
- Validity period: The maximum number of days for which the password is valid. A value of 99999 indicates that the password is still valid.
- Remaining validity period: Number of days before expiry.
- Invalidation duration: Number of days after expiry causing the account to be deactivated. An empty field indicates that there is no deactivation.
- Expiry date: Expressed as the number of days since the reference date (1/1/70).
- Reserved field.

2.3 The file “/etc/group”

This file contains information about the groups present on the system. The syntax of a line in this file is as follows:

```
group:*:GID:users
```

- *group*: the name of the group,
- *: can remain empty or contain the character * or x,
- *GID*: unique identifier for the group in the form of a numeric value,
- *users*: contains the list of users belonging to the group. These users are separated by commas.

3. Useful commands for information about groups and users

3.1 Command “cut”

The command “*cut*” is one of the text filtering tools available in Linux. It is used to extract specific columns from lines of text.

The delimiter is a space, tab, comma, colon (:) or any other character used to separate words in a line. For example, a colon is used as a delimiter in the file “/etc/passwd” to separate different values. Its syntax is :

```
cut options [file]
```

Some cut options:

- f (- fields) Specify the fields you want to extract.
- c (- characters) Specify the characters you want to extract.
- b (-bytes) Specify the bytes you want to extract.
- d (-Delimiter) Here you specify the delimiter you want to use with the *command cut*.

By default, the “tab” is considered to be a delimiter.

Example: the file "*student.txt*" :

```
S101,Ali,91

S102,Mohamed,84

S103,Leila,989
```

```
cut -d ',' -f 2 student.txt
```

```
Ali

Mohamed

Leila
```

3.2 Command "uniq"

The command "*uniq*" is a very powerful command, used in conjunction with the command "*sort*", particularly for analyzing log files. It can be used to sort and display entries by removing duplicates.

Example:

Consider the following file:

```
antoine
xavier
patrick
xavier
antoine
antoine
```

```
sort prenoms.txt | uniq
```

```
antoine
patrick
xavier
```

The command *uniq* can be used to remove duplicates when used without an option.

To display lines that appear only once, we use the ***-u*** option.

Conversely, to display only lines that appear at least twice in the file, use the ***-d*** option:

```
sort prenoms.txt | uniq -d
```

```
antoine
xavier
```

3.3 Command “awk”

“awk” is a very powerful command that searches for strings and performs actions on the selected lines. It is useful for retrieving information, generating reports and transforming data, among other things. The syntax of awk is as follows:

```
awk [-F] [-v var=valeur] 'program' file
```

or

```
awk [-F] [-v var=value] -f file-config file
```

- The argument “-F” must be followed by the field separator (-F: for a ":" field separator).
- The argument “-f” followed by the name of the awk configuration file.
- The argument “-v” defines a variable (var in the example) which will be used later in the program.

An awk program has the following structure: ***string selection criterion {action}***, when there is no criterion, it means that the action applies to all the lines in the file.

Example:

```
awk -F":" '{print $NF}' /etc/passwd
```

There are no criteria, so the action applies to all lines in the /etc/passwd file. The action consists of displaying the number of fields in the file. NF is an awk predefined variable, equal to the number of fields in a line.

4. Commands for user management

4.1 Add a user to the system

The command used to create a new user or update their information is ***useradd***, whose syntax is as follows:

```
sudo useradd [options] login
```

The main options :

- “-c”: displays information about the user (name, position, etc.).
- “-d”: displays the path to the user's home directory.
- “-D”: displays the default options. You can change their values using options attached to the *useradd* - D command.
- “-e”: displays the account expiry date. The format is YYYY-MM-DD.
- “-f”: displays the number of days following password expiry after which the account is deactivated. A value of 0 deactivates the account as soon as the password expires. A value of -1 (default) disables this feature.
- “-g”: displays the name or number of the user's initial connection group. The group name or number must exist. The default group number is 1.
- “-G”: displays the other groups (separated by commas) to which the user belongs.
- “-m”: the user's directory will be created (this is not done by default).
- “-p”: enter the optional password.
- “-s”: indicates the shell launched when the user logs on.
- “-u”: displays the user's unique identifier.

Example 1: you want to create the user ***user1*** with the following options :

- include the comment: user user1,
- define his shell: /bin/bash.
- create his home directory,
- define his home directory: /home/user1.
- make sure it belongs to the Students and Engineers groups.
- make sure that the account is deactivated immediately after the password expires.

```
sudo useradd -c "user user1" -f 0 -G Students,Engineers -m -s /bin/bash user1
```

Example 2: you want to create the user ***user2*** with the following options:

Create the user ***user2*** belonging to the group "***teachers***" (which must exist), with the password is "***info-se***", the home directory is "***/home/user2***" and the shell is "***/bin/bash***":

```
sudo useradd -g teachers -p "info-se" -d /home/user2 -s /bin/sh user2
```

When created by default, the account has no password and is locked. A password must be assigned to unlock the account.

4.2 Delete a user

The command ***userdel*** is used to delete a user's account. The syntax of this command is :

```
sudo userdel [options] login
```

The only option is "***-r***", which deletes the user's home directory.

Example: `sudo userdel -r user1`

4.3 Change password

To change the password, simply type the following command:

```
sudo passwd [options] login
```

Some options:

- "*-d*": delete the password. This operation is performed by the root user.
- "*- k*": only the password is updated, without affecting the expiry properties.

4.4 Display some information about users

To find out the identity of the current user, use the command: ***whoami.*** This command (***whoami***) displays the current user's login.

The commands "***who***", "***users***" and "***w***" are used to find out which users are currently connected to the machine.

4.5 Modify a user account

If a user account has already been created, you can modify it. The command takes the following form:

```
sudo usermod [options] login
```

You can use the same options as those used with the command "***useradd***".

Example 1:

```
sudo usermod -d /home/user2 -m user1
```

This changes user1's home directory to /home/user2. This command also copies the contents of the old home directory and adjusts the rights.

Example 2:

```
sudo usermod -l user2 user1
```

This changes the name of the user user1 to user2. For this to work, user1 must not be connected to the machine when you type the command.

For the changes to take effect, the user affected by the changes must log out and then log back in.

4.6 Change the unique administrator account ID

Changes cannot be made while the session is active. It is therefore necessary to restart the computer in recovery mode and then open a session in super-user mode.

The commands to enter are as follows:

```
usermod --login new_id --home /home/new_id --move-home old_id
exit
```

If you get the error 'unable to lock /etc/passwd', the files are probably read-only. In this case, run

```
mount -o remount,rw /
```

and enter the previous command again.

Then choose <resume> to continue loading Ubuntu normally.

5. Commands for managing groups

5.1 Create a group

The command is as follows:

```
sudo groupadd [option] group
```

The options of the command "*groupadd*" are :

- "-f": Stops the command if the group or GID of the new group already exists.
- "-g": Allows you to choose the numeric value of the new group's GID. This identifier must be unique.
- "-r": Used to add a system group whose GID is less than 500.

5.2 Delete a group

The command is in the form :

```
sudo groupdel group
      or
sudo groupdel GID
```

You cannot delete a group if it is a user's main group.

5.3 Display information about groups

To find out which groups a user belongs to, use the command "***groups***" without any arguments. This command displays the groups of the current user and if you want to find out the groups of another user, simply pass the user's login as an argument to the command "***groups***".

```
$groups
```

This command returns the groups of the current user.

```
$groups user1
```

This command returns the groups of user1.

The command "***id***" is used to find out which groups are active:

```
$id
```

5.4 Modify an account's secondary groups

The command is: ***$ usermod***

This command supports the same options as the command "*useradd*".

Example:

```
$sudo usermod -G toto,users,Students,Engineers toto
```

This command adds the user "toto" to the groups "users", "Students", "Engineers".

You can also add or remove users from a group:

"***$gpasswd -a***" to add a user.

"***$gpasswd -d***" to delete a user.

5.5 List the members of a group

No special command allows this, but you can do it by typing for example:

```
$grep -e '^teachers:' /etc/group | cut -d: -f 4
```

This line will tell you who the users belonging to the group "*teachers*".

6. Changing the identity

6.1 Changing session

To log in under a user's name, you can log out and log in again under this new user.

6.2 In the current session: su

If you just want to temporarily change your identity in a terminal, you can use the su command with the user name. Once it has been validated, all you have to do is type in the user's password to identify yourself under that name.

```
$ su user
```

Exercises

Exercise n°1

1) In which file is your user account defined?
2) How can you check your identity and the group(s) to which you belong?
3) Display the line in the file "*/etc/passwd*" that contains your account details.
4) Display the list of user connection directories declared in the file *"/etc/passwd"*.
5) Display the number of users using the bash "shell".
6) Remember that each user has a shell associated with them, which is launched when they log in. The corresponding shell is indicated in the 7th field of the "*/etc/passwd*" file. In one command line, display the number of different shells listed in "*/etc/passwd*".

Exercise n°2

Identify the commands used to :

1) Find out whether the user "*user1*" is logged in.
2) Display the number of users on the system.
3) Display the list of users in alphabetical order.
4) Find out how many processes "*user1*" has.
5) Find out the number of "*root*" processes.

Exercise n°3

1) Create two groups "*group1*" and "*group2*"?
2) Create four users "user1", "user2", "user3" and "user4"?
3) Put the users in the groups:
   - The first and second users are members of "*group1*".
   - The third and fourth users are members of "*group2*".
   - The second user is also a member of "*group2*".
   - The fourth user is also a member of "*group1*".
4) Check the members of "*group2*" ?

Exercise n°4

Use the appropriate commands to answer these questions:

1) Connect as linux administrator.
2) Display the user configuration files.
3) Display the file *"/etc/passwd*" using three different commands.
4) We want to display the identity of the user "*user*".
5) Establish the user "*user11*" with the following conditions:
   a) its home directory *"/home/toto1*" (create it first)
   b) its id 508
   c) its GId 601 for the group rtoto (create the group first)
   d) its password is "*0/toto1*"
6) Change its password to "*0/guest1*".
7) Display the line relating to "user11" in the file *"/etc/shadow*".