Lab 6

Task Scheduling

6.1 Objectives

Be able to schedule tasks at specific and periodic times and dates

6.2 Before you start

For this session you should have the code of the scripts requested in a previous session:

- `ocupacio.sh`

- `badusers.sh`

6.3 Task execution at an arbitrary time

❓ *What command is used to execute commands at an arbitrary and specific time?*

```
            at
sudo apt
            sudo systemctl enable --now atd
```

Use this command to schedule the following tasks:

- A deletion of files in the */tmp* directory at 2:00h this evening.

- Make a list of users connected to the server after 10 minutes. This listing has to be stored in a file a the home directory of the root superuser. The filename should reflect the date in which the list has been taken.

- A shutdown of your machine at the end of the class. Send a warning to all the connected users five minutes before the shutdown.

❓ *What commands you have used?*   https://phoenixnap.com/kb/linux-at-command

```
    echo "rm -rf /tmp" | at midnight +2 hours        echo "shutdown now" | at end of the class

    echo "who > who.txt" | at now +1 minutes
```

We want to restrict the use of task scheduling for all the users and allow only the superuser to do do.

❓ *Which file you have to create. With contents it should have?*

> crear el fitx /etc/at.deny   i posar els usuaris dins

Try to use the mentioned command with another user and check that it does not allow him/her to do so.
https://www.redeszone.net/tutoriales/servidores/cron-crontab-linux-programar-tareas/

6.4 Task execution at periodic times

Look at the `man page` of the `cron` and `crontab` and answer the following questions:

?   *How you can see what are the contents of a user's `cron` schedule?*

> crontab -u user -l

?   *How you add new schedule for a periodic task?*

> crontab -u user -e

?   *How you can limit who has access to the `cron` service?*

> /etc/cron.deny

6.4.1 Checking the cron schedules

A particular problem that appears when testing `cron` schedules is how to make sure that we have properly specified the frequency without having to wait to the specific time (particularly when is a long time).

?   *How you can verify rapidly that a `crontab` schedule is correct without waiting?*

> with /etc/cron.daily we can force it

?   *What side effects does it have?*

> bugs

6.4.2 Scheduling disk space monitoring

Now we want to monitor the activities of some users (using the script `ocupacio.sh`). The users are listed at a particular file called */etc/ocupacio.users*. The monitoring should be executed every Sunday.

What entry you have added to the `crontab`?

```
0 0 * * sunday(0) /etc/ocupacio.sh
```

Which user is required for executing this entry?

```
root
```

6.4.3 Scheduling user monitoring

We also want to monitor the unneeded users in the system using the script `badusers.pl`. This control will be executed the first day of each month.

What entry you have added to the `crontab`?

```
0 0 1 *  * /etc/badusers.pl
```

Which user is required for executing this entry?

```
root
```

What you have to do for sending by email the list of users produced by the script?

```
| mail -s "message" admin@gmail.com
```

6.4.4 Scheduling Backups

Finally, the user `aduran` wants to do backups of his home directory. Use cron for that purpose taking into account the following conditions:

- A full backup at the first Monday of each month.

- An incremental backup every Wednesday and Saturday

- The backups should be stored at the *home* directory of the user aduran. Obviously don not perform backups of the backups.

- The name of the backup must include the date in which it has been made and the type of backup.

- To reduce the space occupied by the backups, the first day of each month it will be checked if total backup size is greater than 100 MB. If that's the case, you have to delete some backups, as many backups as necessary to be on the space limit. Remove the backups in chronological ascending order (oldest first).

- The user aduran only wants to receive messages if there are errors.

　　?　　*Explain how you have implemented all the above backup scheduling*

Bibliography