3.17 etc.

Como hemos visto con anterioridad, nuestra Raspberry Pi es un ordenador a todos los efectos y en el mismo tenemos un sistema operativo completo con lo que podemos llevar a cabo cualquier tarea que pueda automatizarse.

Por ejemplo, en mi caso, tengo una raspberry pi 2 model B con raspbian que cada noche a una determinada hora se conecta al servidor web dónde alojo el recurso didáctico trivinet.com y copia a la SD de esta raspberry Pi la última copia de seguridad que he hecho en ese servidor web. De este modo tengo la copia de seguridad en 2 ubicaciones distintas por si ocurriese algún desastre.

¿cómo programar tareas para que se ejecuten en fechas/concretas?

En sistemas operativos linux tenemos cron mientras que en sistemas operativos windows existe una herramienta llamada tareas programadas. En este curso nos centraremos en cómo hacerlo en Rasberry Pi os (anteriormente llamado Rasbian).

Para ver el contenido del fichero cron ejecutaremos el comando `crontab -l` mientras que para editar su contenido ejecutaremos `crontab -e` Pongo a continuación una captura de pantalla del resultado de ejecutar el primer comando visto en la máquina antes mencionada

Copyright 2026

```
pi@raspberrypibackups:~ $ crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
0 5 * * * /home/pi/scripts/copiaSeguridad.sh
```

En la imagen anterior podemos ver varias líneas que comienzan por # estas líneas son comentarios, es decir, cron las ignora. Si nos fijamos en la única línea que tenemos que no comienza por # tenemos lo siguiente

`0 5 * * * /home/pi/scripts/copiaSeguridad.sh`

¿y esto qué significa Pablo? pues que siempre que la fecha y hora sea:

- 0 minutos
- 5 horas
- de cualquier día (numérico)
- de cualquier mes
- de cualquier día de la semana (lunes a domingo)

se ejecutará el comando `/home/pi/scripts/copiaSeguridad.sh` Es decir, yo ahí estoy automatizando el que este comando se ejecute siempre a las 5 de la mañana.

En el siguiente enlace https://geekflare.com/es/crontab-linux-with-real-time-examples-and-tools/ podemos ver mas información sobre el uso de cron

Copyright 2026

Una vez que sabemos como programar tareas "únicamente" quedaría la creación del script cuaestión esta que requeriría de un curso por si solo.

> Te animo a que si tienes un script que crees que puede resultar útil a cualquier compañero/a nos lo facilites a fin de recopilarlo en esta sección. Si por el contrario tienes la necesidad de un script para una tarea concreta háznoslo saber y trataremos de darle solución y publicarlo aquí

Revision #4
Created 2023-02-04 10:07:36 CET by Pablo Ruiz
Updated 2023-03-18 10:16:45 CET by Pablo Ruiz

Copyright 2026