



CRONTAB Y RESPALDO

Héctor Andrey Hernández Alonso
Ingeniería en Informática
249757
201501400580
Administración de Sistemas
240003
18 de octubre 2018

Resumen

El programa de utilidad de software cron es un programador de trabajos basado en el tiempo en sistemas operativos de computadora tipo Unix. Las personas que configuran y mantienen entornos de software utilizan cron para programar trabajos (comandos o scripts de shell) para que se ejecuten periódicamente a horas fijas, fechas o intervalos.

Cron está dirigido por un archivo crontab (tabla cron), un archivo de configuración que especifica los comandos de shell para ejecutarse periódicamente en un programa determinado. Los archivos crontab se almacenan donde se guardan las listas de trabajos y otras instrucciones para el daemon cron. Los usuarios pueden tener sus propios archivos crontab individuales y, a menudo, hay un archivo crontab en todo el sistema (generalmente en /etc o un subdirectorío de /etc) que solo los administradores del sistema pueden editar.

```
# _____ minuto (0 - 59)
# _____ hora (0 - 23)
# _____ día del mes (1 - 31)
# _____ mes (1 - 12)
# _____ día de la semana (0 - 6) (de domingo a sábado);
# | 7 también es domingo en algunos sistemas)
#
# * * * * * comando para ejecutar
```

Contenido

Para generar un script que nos permita respaldar la información se necesita de dos cosas:

1. El conocimiento de crear un archivo tar.gz para poder guardar toda la información del usuario
2. Conocimiento de ssh para poder enviar a otro usuario el archivo generado

Para poder obtener la fecha actual se usa el comando **date**

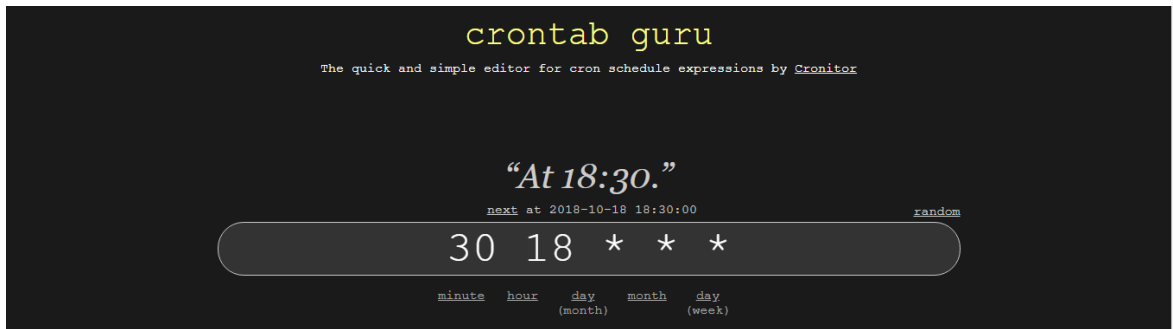
Ejemplo de script

```
GNU nano 2.5.3          Archivo: script_resp.sh          Modificado
#!/bin/bash
fecha=$(date +%Y_%m_%d_%H_%m_%s)
echo $fecha
tar -czuf carpeta_$fecha.tar.gz ../andreyhz/carpeta/
scp carpeta_$fecha.tar.gz oswa@148.224.55.142:./respaldo
echo "enviado"
```

Pantalla de Crontab para establecer los tiempos

```
GNU nano 2.5.3          Archivo: /tmp/crontab.CpeBpo/crontab
# 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
* * * * * /home/andreyhz/cron/scriptHora.sh
30 18 * * * /home/andreyhz/script_resp.sh
```

Página de ayuda para poder establecer los parámetros para el crontab



Descripción de contenido:

- **#!/bin/bash**
 - Permite crear el script
- **fecha=\$(date +%Y_%m_%d_%H_%s)**
 - Guarda en la variable fecha la fecha actual
- **tar -czvf respaldo.tar.gz ../andreyhz/**
 - Genera un archivo comprimido .tar.gz
- **scp respaldo.tar.gz oswa@148.224.55.142:/respaldo**
 - Se envía al usuario

Conclusión

Gracias a Crontab podemos establecer la acción de un script que se realice en determinado tiempo sin que nosotros tengamos que hacerlo en ese momento.

Bibliografías

<https://docs.bluehosting.cl/tutoriales/servidores/como-hacer-un-respaldo-de-sus-archivos-a-traves-de-ssh.html>

<https://searchsecurity.techtarget.com/definition/Secure-Shell>

https://crontab.guru/#30_18_*_*_*