Automatización de tareas

Sumario

- 1 Automatizando Tareas con cron
  - 1.1 Sintaxis de crontab
  - 1.2 Ejemplos
  - 1.3 Nota
- 2 El comando at
- 3 Referencias

Automatizando Tareas con cron

cron es una utilidad estándar en Linux utilizada para automatizar tareas que deben ejecutarse periódicamente. En todos los Sistemas Linux se ejecuta un daemon, crond, que se activa una vez por minuto y ejecuta las tareas que tenga pendientes. De este modo mediante un fichero de texto, crontab (tabla de tareas cron), se pueden automatizar tareas especificando su período. La idea es que en un fichero de definición de tareas, crontab, se incluya en cada línea de ese archivo la tarea a ejecutar, como un comando de shell, y el instante temporal en que se va a lanzar. El comando que permite manejar esta utilidad es **crontab**, cuyas principales opciones son:

- -e: Edita la tabla cron del usuario actual
- -l: Lista las tareas cron del usuario actual
- -u: Especifica el usuario del que queremos administrar su crontab (solo root puede invocar esta opción)
- -r: Elimina la tabla de cron del usuario

También se pueden definir tareas periódicas editando, como root, directamente el archivo /**etc**/**crontab**. Otra alternativa es utilizar los directorio /**etc**/**cron.daily** (ejecución diaria), /**etc**/**cron.weekly** (semanal) y /**etc**/**cron.monthly** (mensual), dentro ode los cuales se ubicarán scripts que deberán ser ejecutados con la periodicidad indicada y según lo establecido en /**etc**/**crontab**

Sintaxis de crontab

El formato de configuración de cron es muy sencillo:

- El símbolo almohadilla «*» es un comentario, todo lo que se encuentre después de ese carácter no será ejecutado por cron.
- El momento de ejecución se especifica de acuerdo con la siguiente tabla:
  - Minutos: (0-59)
  - Horas: (0-23)
  - Días: (1-31)
  - Mes: (1-12)
  - Día de la semana: (0-6), siendo 1=lunes, 2=martes,... 6=sábado y 0=domingo (a veces también 7=domingo)

algunos sistemas soportan la especificación de divisores del tiempo a la hora de definir la tarea. Por ejemplo:

```
/2: cada 2 minutos
/10: cada 10 minutos
```

Ejemplos

30 10 1 /usr/bin/who >> /home/quien.tex

Ejecuta la orden who todos los lunes a las 10:30 (campo primero, minutos, 30, campo segundo, hora, 10, último campo, día de la semana, 1, lunes) y guarda la salida en el fichero quien.tex

30 21 6 /sbin/shutdown -h now

Ejecuta un apagado de la máquina todos los sábados a las 21:30

0 12 find /tmp -ls -exec rm -r {} \;

Elimina todos los archivos temporales en /tmp todos los días a las 12 del mediodía

Nota

Cada vez que se ejecuta una tarea del cron se envía un mail al usuario que ejecuta la tarea. Para deshabilitar esta característica podemos añadir **>/dev/null 2>&1** al final de la tarea o bien añadir en el archivo del cron la variable **MAILTO=""**

El comando at

Este comando permite ejecutar una tarea programada para un instante de tiempo dado. Al contrario que cron, no se utiliza para programar de modo persistente las tareas, sino que está más indicada para tareas puntuales que deben de ser ejecutadas con cierta demora. Veamos un ejemplo

```
at 12:00
warning: commands will be executed using /bin/sh
at> reboot
at> <EOT>
job 2 at Tue Apr 19 12:00:00 2016
```

Escribimos el comando, at 12:00, en este caso indicando que la tarea se ejecutará a las 12:00, a continuación entramos en un modo edición interactivo, introducimos el comando a realizar y cerramos con CTRL+D, tras lo cual se creará la tarea con un id asociado, en este caso el id 2

Para ver la cola de tareas de at

```
atq
```

Eliminar una tarea de at

```
atrm 2
```

Esta eliminaría la tarea con número 2

Referencias

Unix cron

http://es.wikipedia.org/wiki/Cron_%28Unix%29

Volver

JavierFP 16:31 08 ene 2019 (CET)