

Tareas programadas

Acciones programadas reiterativas

En el sistema es posible ejecutar una tarea en un momento determinado, esto es muy útil si pensamos en realizar respaldos por ejemplo.

El demonio **Cron** que se carga al iniciar el sistema, es el encargado de realizar tareas de forma programada, al levantar lee el archivo `/etc/crontab` y los los script que se encuentra en los directorios: `/etc/cron.d/`, `/etc/cron.hourly/`, `/etc/cron.daily/`, `/etc/cron.weekly/` `/etc/cron.monthly/`. Toma información de ellos y en que momento se deben ejecutar.

Además se pueden agregar tareas programadas a una hora, día determinado, editando el `crontab` del usuario.

Contenido del archivo `/etc/crontab`

```
[root@acer1 etc]# cat crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# For details see man 4 crontabs

# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR
sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
```

Descripción

Campo	Descripción
SHELL	interprete de comandos
PATH	camino a los ejecutables
MAILTO	a quien mandar un mail si ocurre un error
HOME	localización del directorio raiz

Estructura de los directorios pertenecientes a cron:

```
cron.d/
`-- sysstat
cron.daily/
|-- 0anacron
|-- logrotate
|-- makewhatis.cron
|-- slocate.cron
|-- tetex.cron
`-- tmpwatch
cron.hourly/
cron.monthly/
`-- 0anacron
cron.weekly/
|-- 0anacron
`-- makewhatis.cron
```

Nota: El contenido de estos directorios varía según los paquetes instalados en el sistema.

Script en /etc/cron.daily

```
[root@cursosvirtuales cron.daily]# cat 0logwatch
#!/bin/bash
```

```
DailyReport=`grep -e
"^[[:space:]]*DailyReport[[:space:]]*=[[:space:]]*"
/usr/share/logwatch/default.conf/logwatch.conf | head -n1 | sed -e
"s|^\s*DailyReport\s*=\s*||" `
```

```
if [ "$DailyReport" != "No" ] && [ "$DailyReport" != "no" ]
then
    logwatch
fi
```

Script en /etc/cron.d

```
[root@cursosvirtuales cron.d]# cat sysstat
# Run system activity accounting tool every 10 minutes
*/10 * * * * root /usr/lib64/sa/sa1 1 1
# 0 * * * * root /usr/lib64/sa/sa1 600 6 &
# Generate a daily summary of process accounting at 23:53
53 23 * * * root /usr/lib64/sa/sa2 -A
```

Debe observarse la diferencia entre un script en el cron.daily y el cron.d:

En el cron.d debe agregarse: el momento de realizarlo y el usuario. Son los scripts contenidos en este directorio configurables en su exacto momento de ejecución.

Comando crontab

El comando crontab permite configurar tareas programadas personalizadas por un usuario.

Sintaxis: `crontab [-e |-r | -l | -u]`

Opciones	Descripción
-e	Para editar el archivo
-r	Para eliminar
-l	Para ver el contenido
-u	usuario

Ejemplo de la configuración personal del crontab:

```
crontab -l
05 * * * 7 /root/bin/cambio.sh
*/05 * * * * /root/bin/controlaip.sh
01 06 * * 2-6 /root/bin/crear-respaldo.sh
```

Campos del crontab

	Campo	Valores permitidos
T i e m p o	minuto	0-59
	hora	0-23 (00 medianoche), /10 significa cada 10 minutos
	día del mes	1-31
	mes	1-12 (o su nombre en inglés)
	día de la semana	0-7 (0 o 7 es el domingo), o su nombre en inglés
	comando	

Seguridad

Se puede determinar que usuarios pueden utilizar el contrab, y que usuarios no editando el archivo correspondiente:

Usuarios autorizados listados en el archivo: **/etc/cron.allow**

Usuarios no autorizados listados en el archivo: **/etc/cron.deny**

Otras Taréas periódicas

Anacron

Una tarea tipo *cron* es muy útil pero su ejecución depende de que la máquina se encuentre encendida en el momento exacto para el que se programó. Es por ello que existe otra tipo de tareas periódicas: las **anacrons**. Estas se ejecutan cada cierto tiempo especificado en días. En caso de que la máquina se apague durante un tiempo mayor que el especificado, entonces una vez encendida y activado el servicio que manipula los trabajos *anacron* todos aquellos que se encuentren atrasados se ejecutarán tan pronto transcurra la espera especificada. O sea, para cada trabajo *anacron* se indica un tiempo en días y una espera (*delay*). Los trabajos *anacron* se almacenan en el archivo del sistema **/etc/anacrontab**. Además de indicar el intervalo de **días**, el **delay** y el comando se especifican un **nombre para el trabajo**. Este se emplea nombrar un archivo que utiliza el servicio de *anacron* para saber cuando fue la última vez que ejecutó este trabajo (*timestamp file*). Estos archivo se almacena en **/var/spool/anacron/** y sólo contienen una fecha.

```
[root@cursosvirtuales etc]# tree /var/spool/anacron/
/var/spool/anacron/
├── cron.daily
├── cron.monthly
└── cron.weekly
```

Estructura del archivo anacrontab

El archivo **/etc/anacrontab** tiene la siguiente forma:

```
[root@cursosvirtuales etc]# cat anacrontab
# /etc/anacrontab: configuration file for anacron

# See anacron(8) and anacrontab(5) for details.

SHELL=/bin/sh
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
# the maximal random delay added to the base delay of the jobs
RANDOM_DELAY=45
# the jobs will be started during the following hours only
START_HOURS_RANGE=3-22

#period in days   delay in minutes   job-identifier   command
1         5         cron.daily       nice run-parts /etc/cron.daily
7        25         cron.weekly      nice run-parts /etc/cron.weekly
@monthly 45         cron.monthly     nice run-parts /etc/cron.monthly
```

Descripción del /etc/anacron

Campo	Descripción
1 / 7 / @monthly	Día/s
5 / 24 /45	Espera
nice	Prioridad
Run-parts	Indica que se deben ejecutar los scripts que se encuentran en los directorios
/etc/cron.daily	Directorios

Podría pensarse que puede ocurrir que un mismo programa se ejecute por la vía *cron* y por la *anacron*, pero esto no sucede pues existe una tarea *cron* para cada período que actualiza los archivos en los cuales se basa el servicio *anacron* para saber que es lo que debe ejecutar; o sea, cuando el servicio *cron* logra ejecutar las tareas correspondientes a un período, actualiza el archivo de ese período, ubicado en: `/var/spool/anacron/` y cuando llega *anacron* (que se demora más, pues debe esperar un tiempo) ya no tiene que ejecutar las tareas.

Para arrancar el servicio *anacron*, use el comando `/sbin/service anacron start`. Para detener el servicio, use el comando `/sbin/service anacron stop`. Se recomienda arrancar el servicio en el momento del arranque.