

Administración Linux

Servicios en Linux

MSc. Ivan A. Escobar

<http://ivanescobar.com>

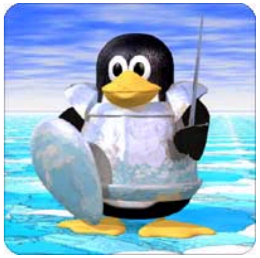
iescobar@itesm.mx

Automatización tareas: cron



- Sistema que permite a los usuarios crear trabajos que corre en un determinado momento.
- Cada usuario cuenta con su propio crontab y el sistema en general cuenta con uno que solo puede ser modificado por aquellos con acceso a root.
- Elementos comunes y principales
 - el demonio cron
 - el comando crontab

El demonio cron



- Proceso que se arranca al encender la máquina y está permanentemente activo.
- Su misión es inspeccionar cada minuto los ficheros crontabs de los usuarios y ejecutar los comandos que allí se digan a los intervalos horarios que se indicaron.
- Los crontabs son dependientes de cada usuario.



min hora dia mes dia-semana objeto-ejecución

- Minuto (0-59)
- Hora (0-23)
- Día del mes (1-31)
- Mes del año (1-12)
- Día de la semana (0=Domingo, 6=Sábado)
- Nombre completo del archivo que contiene el proceso shell a ser ejecutado



Notación campos numéricos

- Posible simplificar la notación para los campos numéricos:
 - asterisco “*”: todos valores autorizados
 - en campo hora: ejecución todas las horas
 - valores separados por un guión: “-”: intervalo de tiempo
 - ejemplo: 2-5 en campo mes:
 - todos los meses de febrero a mayo
 - valores separados por comas “,” la acción es realizada para todos esos valores:
 - ejemplo: 5,10,15 en campo minutos y 3 en campo hora:
 - ejecución comando a las 3:05, 3:10 y 3:15
 - posible usar nombres para el mes y día de la semana
 - usar las tres primeras letras del día o la semana

Dos ejemplos de entradas



min hora dia mes dia-semana objeto-ejecución

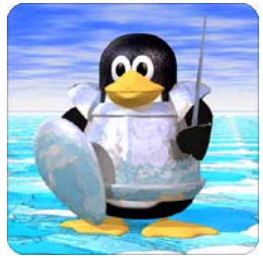
- El siguiente ejemplo ejecutará el script cada minuto, a cada hora, todos los días, todos los meses :

*** * * * * root /usr/TSS/bin/scriptweb**

- Mensualmente el día primero de cada mes a las 3:01 am

01 3 1 * * root /usr/TSS/bin/scriptweb

Ejemplo archivo *crontab* en BSD



```
iescobar@localhost:8> cat /etc/crontab
```

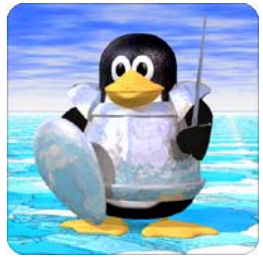
```
SHELL=/bin/bash  
PATH=/sbin:/bin:/usr/sbin:/usr/bin  
MAILTO=root  
HOME=/
```

variables que se usan para configurar el entorno en el que se ejecutan las tareas cron
MAILTO="", no se envia correo alguno
HOME directorio principal que deberá usarse al ejecutar los comandos o scripts

```
# run-parts
```

```
01 * * * * root run-parts /etc/cron.hourly  
02 4 * * * root run-parts /etc/cron.daily  
22 4 * * 0 root run-parts /etc/cron.weekly  
42 4 1 * * root run-parts /etc/cron.monthly
```

Ejemplo archivo crontab



```
45 00 * * 1-5
#55 06, 09, 15, 19 * * 1-5
#00 08 * * 1-5
#55 23 * * 1-5
00 02 * * 1
00 * * * *
00, 05, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55 * * * * root /usr/lib/atrun
00, 05, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55 * * * * root /bin/date > /datesync
00 09 * * 1-5
00 18 * * 1-5
daemon /usr/bin/calendar -
uucp /usr/lib/uucp/uu.hourly -all
uucp /usr/lib/uucp/uu.daily
uucp /usr/lib/uucp/uucp.daily
root /usr/ucb/messages -c 14
root /usr/ucb/logger 'date'
```


Creando archivos crontab



- Dos formas de crear un archivo crontab
- Primera opción
 - Editar en un archivo el conjunto de ordenes a programar.
 - Utilizar el comando crontab seguido del archivo;
 - se crea archivo en directorio `/var/spool/cron` con el nombre del usuario.
- Segunda opción
 - usar comando crontab
 - `crontab [-u usuario] [opciones]`
 - opciones:
 - `e` : permite editar un archivo crontab para el usuario actual o para otro usuario si hemos especificado otro.
 - `r` : borra un archivo crontab.
 - `l` : lista el contenido del archivo crontab
- Un archivo por usuario ubicado en `/var/spool/cron`
 - root
 - usuario 1
 - usuario 2
 - :
 - usuario n