

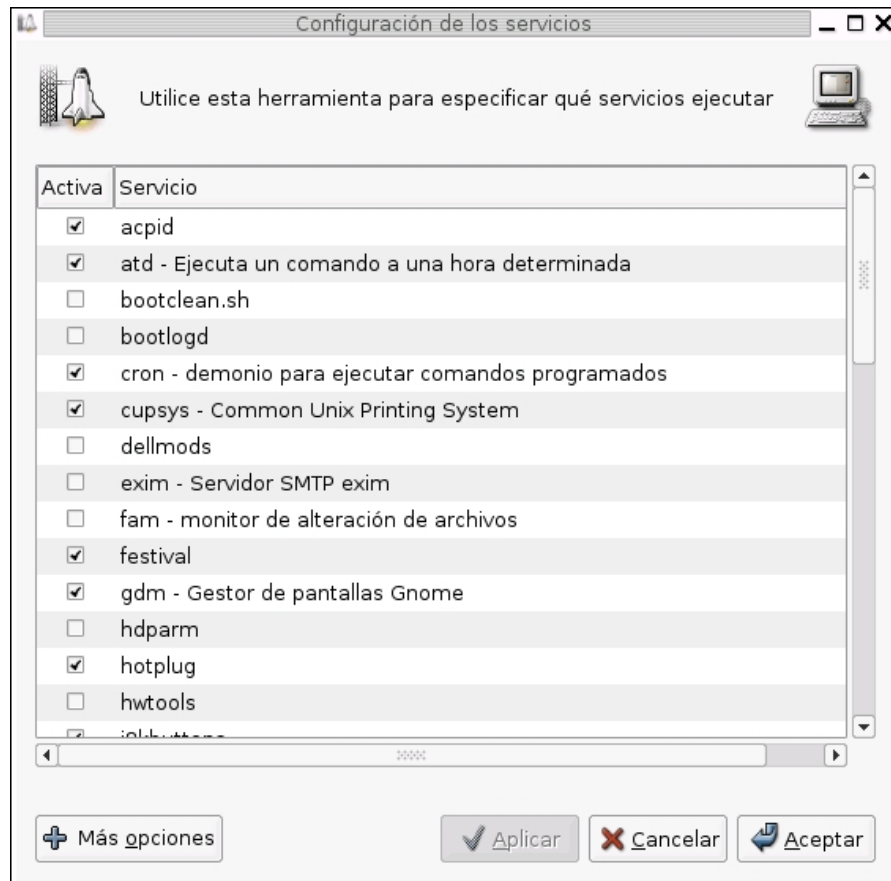
CAPÍTULO 3

Administración básica del sistema

Arranque y Parada de Servicios

Existen en un sistema Linux un conjunto de programas que están en continuo funcionamiento; estos programas son conocidos por demonios o “daemons”, que están asociados a determinados “servicios” y que se encargan de servir a determinados propósitos: los hay que ejecutan una tarea periódicamente, otros almacenan información, y otros simplemente están a la espera de que alguien los necesite (por ejemplo, el servicio de impresión). No todos los servicios tienen un demonio asociado, los hay que únicamente son scripts que ejecutan una serie de tareas (aquellos que lo tienen suelen terminan en una “d” para indicar eso precisamente).

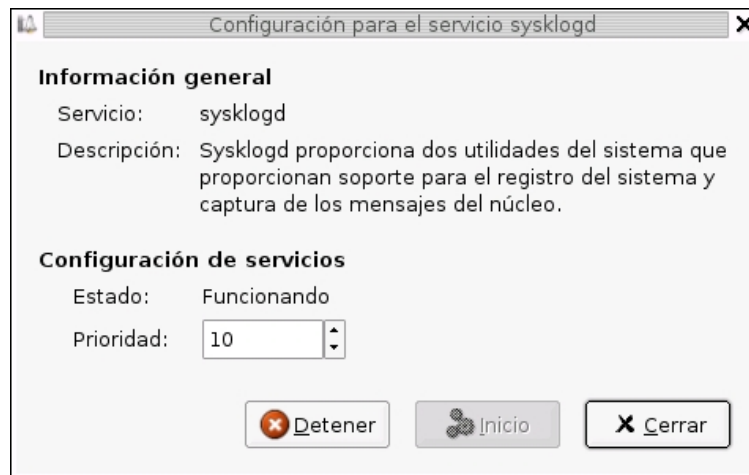
Para poder seleccionar qué servicios queremos tener funcionando en nuestro sistema y cuales disponemos de una aplicación gráfica denominada “Servicios”, situada en el menú: Aplicaciones -> Herramientas de Control -> Panel de Control, cuya imagen puede ver a continuación.



En la lista aparecen todos los servicios disponibles; con una marca se señalan los que están en funcionamiento actualmente. Iniciar o detener un servicio es tan simple como marcar o desmarcar la casilla correspondiente. Ten en cuenta que a medida que vayas instalando ciertos programas la lista irá creciendo.

No conviene tener activos servicios que no vamos a utilizar: tener servicios funcionando que no usamos puede hacer al sistema vulnerable (y puede hacer que funcione más lento), mientras que si detenemos algún servicio esencial el sistema puede funcionar mal.

Como puede que no sepamos qué hace un servicio determinado, disponemos de una pequeña ayuda, para lo que hay que activar el botón “Más opciones”; de esta forma aparece un botón de “Propiedades del servicio”, en el que una ventana emergente nos muestra información relevante de un servicio concreto. En la siguiente imagen se muestra esta ventana para “sysklogd”, un servicio encargado de almacenar los mensajes que emite el núcleo y que pueden servir para identificar un mal funcionamiento del sistema.



Además de proporcionarnos una manera alternativa de iniciar o detener el servicio, podemos establecer su prioridad, es decir, el orden en el que se inician respecto al resto de servicios; la prioridad oscila desde el 1 (el primero en iniciarse) hasta el 99 (el último en hacerlo).

Al mismo tiempo, el botón “Más opciones” enseña un menú donde se pueden seleccionar varias listas, cada una para configurar la lista de servicios a iniciar en diferentes situaciones:

- Deteniendo el sistema: son los scripts que se ejecutarán al apagar la máquina; generalmente desactivan alguna característica o apagan algunos dispositivos (desmontar unidades, por ejemplo)
- Modo gráfico: son los servicios a que se activarán al tener configurado el inicio en modo gráfico (gdm)
- Modo texto: servicios a iniciar al arrancar el sistema en modo texto (consola)
- Reiniciando el sistema: igual que el primero, pero cuando se va a reiniciar el sistema.

Puede parecer extraño la división anterior, pero la razón hay que buscarla en los sistemas Unix tradicionales, en los que es necesario distinguir entre un reinicio y un apagado total del sistema; para ordenadores personales ambas situaciones son equivalentes y no presenta más inconveniente que tener que configurar dos veces las mismas opciones. Del mismo modo hay servicios que no se necesitan en caso de iniciar en modo texto (como por ejemplo el demonio “gdm”).

Ejecución periódica de comandos

Algunas veces surge la necesidad de ejecutar un programa de manera regular a intervalos periódicos, o que puedan ejecutarse en determinados momentos sin tener que estar delante del ordenador (por ejemplo, para realizar copias de seguridad a altas horas de la noche, cuando nadie esté usándolo). Afortunadamente Linux proporciona las herramientas necesarias para llevar a cabo estos dos tipos de tareas: `at` y `cron`, asociados a dos demonios; `atd` y `cron` respectivamente, que deben estar en funcionamiento para que esto sea posible. No es necesario reiniciar estos servicios tras realizar algún cambio porque ellos mismos comprueban periódicamente si tienen tareas a realizar.

`at`

Este comando permite utilizar la línea de comandos para configurar un evento individual. Tanto el administrador del sistema como los usuarios tienen acceso a esta herramienta.

El comando "`at`" dispone de una jerarquía de colas; cada una de estas colas se refiere a una prioridad o la cantidad de tiempo de procesamiento de la máquina que se le permite ocupar a ese trabajo. Los niveles están nombrados con un único carácter, que puede ir desde la "`a`" hasta la "`z`", o desde la "`A`" a la "`Z`". Cuanto más avanzada sea la letra en minúscula desde el principio del abecedario, menos tiempo se le permite ocupar. Si se utilizan las mayúsculas ocurre lo mismo, pero sólo se ejecutarán si la media de la carga de la CPU es inferior a 0.8 (al 80%).

Para configurar quién puede hacer uso de "`at`" se utilizan dos archivos; `at.allow` y `at.deny`; el funcionamiento es el siguiente: si existe `at.allow`, sólo los usuarios cuyos nombres se encuentren en él podrán utilizar el comando (el superusuario siempre podrá usarlo); si no existe, se comprueba `at.deny`, que designa los usuarios que no pueden utilizar "`at`". Los nombres de usuario deben aparecer cada uno en una línea.

Para utilizar `at` la sintaxis es: `at` opciones fecha.

Algunas de las opciones disponibles son:

- c envía los comandos a la salida estándar.
- f coge la entrada del archivo especificado.
- m envía correo electrónico al usuario original cuando el trabajo se ha completado.
- q asigna una cola específica; la cola "=" se refiere a los trabajos que se están ejecutando en ese momento.
- V proporciona el número de versión de `at`.

Existen muchas formas de decirle a at cuándo queremos que ejecute un comando; las más usadas sin embargo son:

hh:mm hora y minuto
mmddyy mes, día y año; pueden ir separados por "/" o por "."

Ambas pueden combinarse para especificar una fecha completa.

Por ejemplo para crear un trabajo con prioridad "c" a las 15:30 del 15 de Agosto de 2004 debería ponerse:

```
at -q c 15:30 08/15/2004
```

Existe otra opción muy útil, "now + X unidades", que cuenta a partir de ahora X unidades (minutes, hours, days). Para ejecutar el comando dentro de 5 minutos:

```
at now + 5 minutes
```

Una vez puestas las opciones entraríamos en la línea de comando propia de at; todo lo que se escriba ahora y hasta que se pulse "Control-D" se considera parte del trabajo. Hay que tener en cuenta que sólo hay que poner comandos de consola, ya que las órdenes se ejecutarán con el intérprete de comandos /bin/sh.

Para ver los trabajos que están en la cola se utiliza atq; en primer lugar aparece el número de trabajo, seguido de la fecha y hora de ejecución, el nombre de la cola y el usuario a quien pertenece el trabajo.

Si se quiere eliminar un trabajo antes de que se lance, hay que utilizar atrm número_de_trabajo, donde éste número es el que muestra atq.

cron

Se utiliza para gestionar comandos que se ejecutan más de una vez; es una herramienta más completa que at y la forma de utilizarlo depende del usuario que lo haga. Veremos primeramente cómo lo usa root.

El archivo principal de cron es /etc/crontab; en él se define cuándo se ejecutan las tareas, teniendo en cuenta que se pueden configurar cada hora (hourly), día (daily), semana (weekly) o mes (monthly). Cada una de ellas posee un directorio propio denominado cron.periodo, siendo periodo uno de los anteriores. En estos directorios se encuentran los scripts que serán ejecutados.

El comando cron tiene una forma particular de representar el tiempo,

distinta de la de at. Una sentencia de tiempo cron se divide en 5 partes: A H D M W; cada uno de estos elementos se representan numéricamente. Su significado es:

- A Especifica el número de minutos después de la hora; puede valer de 0 a 59.
- H Especifica la hora, en formato de 24 horas; vale de 0 a 23.
- D Especifica la fecha (día del mes); vale de 0 a 31.
- M Especifica el mes del año; vale de 1 a 12 o las 3 primeras letras del nombre del mes.
- W Especifica el día de la semana; vale de 0 (domingo) a 6 (sábado) o las tres primeras letras del nombre del día.

También se pueden utilizar unos modificadores; existen diferentes operadores que puede utilizar para expresar rangos y otros elementos que le permiten expresar valores de tiempo múltiples para su trabajo de cron:

- * expresa que todos los valores son posibles
- especifica un rango de valores (incluyendo a los que aparecen
- , valores individuales
- / especifica el incremento que debería seguir un grupo de valores en lugar de aumentar en uno

Puedes definir tu propia periodicidad, para lo que deberás crear un directorio, aunque realmente no es necesario ya que las 4 entradas que hay en crontab cubren prácticamente todas las necesidades.

Para crear un trabajo de cron como usuario (incluido el superusuario cuando el trabajo no es para el sistema), hay que ejecutar el comando:

`crontab -e`

Así abriremos el archivo de datos crontab para el usuario (si es la primera vez estará vacío); el formato de éste archivo es similar al que vimos anteriormente, salvo que no hay que especificar el nombre de usuario (ya que cada usuario posee el suyo propio).

Si hubo algún error o apareció alguna información al ejecutar un comando, tanto con at como con cron, puede consultarse revisando el correo del usuario.

Ejercicios

- 1) Seguramente tengas una impresora funcionando en tu sistema; prueba a detener el servicio llamado "cupsys" e intenta imprimir cualquier cosa; hasta que no actives de nuevo el servicio no podrás hacerlo.
- 2) Activa si no lo están ya los servicios atd y crond, para poder realizar los ejercicios siguientes.
- 3) Realiza una copia de seguridad programada para dentro de 5 minutos del archivo /etc/hostname en el directorio de root.
- 4) Observa que está programado correctamente.
- 5) Programa una copia del archivo /etc/hostname como antes pero para que se realice cada viernes a las 23:45.
- 6) Como usuario alumno crea una copia de seguridad programada de tu carpeta "Documents" cada 13 de cada mes a las 8:30 de la mañana; el comando es "tar cfz copia.tar.gz Documents"

Soluciones

- 3) El comando es: at now +5 minutes; una vez dentro de la interfaz la orden es:
cp /etc/hostname /root.
- 4) Para ver los comandos programados: atq, y observa
- 5) Introduce al final del archivo /etc/crontab la línea:
45 23 * * 6 root cp /etc/hostname /root
- 6) Ejecuta "crontab -e" y escribe la línea:
30 8 13 * * tar cfz copia.tar.gz Documents