En Hora mini-COMO

Ron Bean, `rbean@execp.com`
Traducido por Rodolfo Pilas, `rodolfo@linux.org.uy`

Cómo configurar y mantener el reloj de su computadora en hora.

Índice General

1 Introducción

Los chips de reloj-de-tiempo-real (*real-time-clock*) que son usados en las placas madres de los PC (y también en las costosas estaciones de trabajo) son notoriamente inexactos. Linux dispone de una forma simple de corregir esto por software, haciendo que el reloj sea potencialmente **muy** exacto sin necesidad de una fuente externa de confirmación del tiempo. Pero muchas personas no reparan en esto, por varias razones:

1. No es mencionado en la mayoría de la documentación sobre "Cómo configurar Linux", y es bastante difícil de configurar automáticamente en el momento de la instalación (sin embargo, sería posible en teoría, si se dispusiera de un módem).
2. Si revisa "`man clock`" verá `clock(3)`, que no es lo que se quiere. (pruebe "`man 8 clock`").
3. La mayoría de las personas no prestan mucha atención a la hora.
4. Aquellos pocos que sí lo hacen, buscan usar el paquete `xntpd` de `http://louie.udel.edu` para sincronizar con una fuente externa el reloj, como un servidor de hora de red o un radio-reloj.

Este mini-COMO describe un acercamiento sencillo al tema. Si Usted está completamente interesado en esto, le recomiendo encarecidamente que invierta un tiempo en `http://www.eecis.udel.edu/~ntp/` donde encontrará todo lo relativo a este interesante asunto, incluyendo información completa sobre `xntpd` y enlaces para NIST y USNO (hay más comentarios sobre `xntpd` al finalizar).

Nota:

> Si usa más de un sistema operativo en su máquina, debe dejar que solamente uno de ellos corrija el reloj del CMOS, para evitar que uno se confunda con otro. Si regularmente corre Linux y Windows en la misma máquina, puede echar una mirada a algún programa de reloj shareware que este disponible para Windows (siga los enlaces del url indicado arriba).

2 Uso del programa `clock`

Todo lo que necesita saber está en la página man de `clock(8)`, y este mini-COMO lo guiará a través del proceso.

Nota:

> Debe ser root para ejecutar "`clock`", o cualquier otro programa que afecte la hora del sistema o el reloj del CMOS.

2.1 Revisión de su sistema

Revise sus archivos de inicio del sistema buscando una línea de comando semejante a "`clock -a`" o "`clock -ua`". Dependiendo de la distribución que esté utilizando, puede estar en `/etc/rc.local` o en `/etc/rc.d/rc.sysinit`, `/etc/rc.d/init.d` o algún otro lugar similar.

Si aparece como "`clock -s`" o "`clock -us`", cambie la "`s`" por una "`a`", y compruebe si tiene el archivo `/etc/adjtime`, que contiene una sola línea semejante a esta:

```
0.000000 842214901 0.000000
```

Estos números son el factor de corrección (en segundos por día), el momento que reloj fue corregido por última vez (en segundos desde Enero 1, 1970), y el momento (segundo) en que fue corregido por última vez. Si no dispone de este archivo, entre en el sistema como root y créelo, con una única línea que como esta (todo en cero):

```
0.0 0 0.0
```

Ejecute "`clock -a`" o "`clock -ua`" manualmente desde la línea de comando para actualizar el segundo número (use la "`u`" si su reloj esta marcando la hora Universal en lugar el tiempo local).

2.2 Cálculo de las variaciones de su reloj

Primero, necesita saber qué hora es `:-)`. Su hora del día puede estar o no ajustada. Mi método favorito es llamar al servicio WWV al (303) 499-7111 (es una llamada por voz). Si dispone de acceso a un servidor de horario en la red, puede utilizar el programa `ntpdate` del paquete `xntpd` (utilice el parámetro `-b` para permitir que el kernel actualice el reloj del CMOS). De lo contrario use `date -s hh:mm:ss` para poner en hora el reloj del kernel a mano, y entonces `clock -w` para el reloj del CMOS, tomando la hora del reloj del kernel. Deberá recordar cuándo puso en hora el reloj por última vez, lo mejor es escribir la fecha en algún lugar donde no la pierda. Si utilizó `ntpdate`, con `date +%s`" puede registrar el número de segundos desde Enero 1, 1970.

Entonces, vuelva algunos días o semanas después para ver cuánto se ha desviado su reloj. Si puso la hora a mano, le recomiendo espere al menos dos semanas para poder calcular la desviación lo más aproximada a .1 seg/día. Después de algunos meses le será posible determinarla con exactitud cercana a .01 seg/día (algunas personas prefieren más exactitud aún, pero yo ya soy conservador a estas alturas). Si ha utilizado `ntpdate` no es necesario que espere tanto tiempo, pero siempre podrá hacer un mejor ajuste cuanto más tiempo pase.

Puede ejecutar con cron el comando "`clock -a`" a intervalos regulares para mantener la hora del sistema ajustada con el horario (corregido) de la CMOS. Este comando puede ejecutarse también desde algún archivo de inicio en el momento del arranque, si Vd hace esto (como lo hacemos casi todos nosotros) será suficiente para sus propósitos.

Tenga en cuenta que algunos programas pueden quejarse si el sistema salta más de un segundo a la vez, o si va para atrás. Si tiene este problema, puede utilizar `xntpd` o `ntpdate` para corregir el tiempo más gradualmente.

2.3 Ejemplo

2.3.1 Estableciendo la hora

Entre como root. Llame al (303) 499-711 (voz), escuche el anuncio de la hora y escriba:

```
date -s hh:mm:ss
```

pero no presione ENTER hasta no escuchar el beep (puede utilizar `ntpdate` aquí en lugar de `date`, y evitar la llamada por teléfono. Esto establece la hora del núcleo (*kernel time*). Luego escriba:

```
clock -w
```

Esto pone en hora la CMOS a partir de la hora del núcleo. Y ahora escriba:

```
date +%j
```

(o `date +%s` si ha usado `ntpdate` en lugar de `date` arriba) y escriba el resultado, para la próxima vez.

2.3.2 Restablecimiento de la hora y cálculo de la desviación.

Busque la fecha en que ajustó la hora la vez anterior. Entre como root y escriba:

```
clock -a
```

Esto hace que el núcleo tome la hora de la CMOS. Llame al (303) 499-7111 (voz) y escuche el anuncio. Entonces escriba:

```
date
```

Presionando ENTER en el momento que escuche el beep, pero mientras está esperando, apunte la hora que escucha y no cuelgue aún. Lo que escucha es la hora que debería tener su máquina cuando se encuentra exacta en el minuto. Escriba:

```
date hh:mm:00
```

usando el minuto **siguiente** al cual acaba de escuchar y presionando ENTER en el momento en que escuche el beep nuevamente (ahora puede colgar). Para `hh` utilice su hora local. Esto coloca la hora del núcleo (*kernel time*). Teclee:

```
clock -w
```

el cual establecerá la nueva hora (correcta) en el reloj del CMOS. Escriba:

```
date +%j
```

(o `date +%s` si es lo que ha utilizado anteriormente)

Ahora dispone de tres números (dos fechas y un intervalo de tiempo) que le permitirán calcular el desvío horario.

2.3.3 Cálculo del factor de corrección

¿Cuándo ejecutó `date` en el minuto, su máquina iba atrasada o adelantada? Si iba adelantada, deberá descontar algunos segundos, aunque le resulte un número negativo. Si estaba retrasada, deberá agregar algunos segundos, aunque resulte un número positivo.

Ahora reste ambas fechas. Si usó "`date +%j`" , el número representando el día del año (1-365 o 1-366 en año bisiesto). Si Usted ha pasado el 1 de Enero entre la primera y segunda fecha entonces deberá agregar 365 (o 366) al segundo número. Si usó `date +%s` su número está en segundos y deberá dividirlo por 86400 para obtener días.

Si ya tiene un factor de corrección en `/etc/adjtime`, deberá tomar en consideración el número de segundos que ha corregido. Si ha corregido adelantando, este número tendrá el signo opuesto al del que ha medido; si ha corregido atrasando tendrá el mismo signo. Multiplique el antiguo factor de corrección por el número de días, y añada el número de segundos resultante (adición con signos - si ambos números tienen el mismo signo: obtendrá un número mayor; si tienen signos opuestos, obtendrá un número menor).

Divida el número total de segundos por el número de días para tener el nuevo factor de corrección, y colóquelo en `/etc/adjtime` en lugar del anterior. Registre la nueva fecha (en segundos o días) para la próxima vez.

He aquí mi `/etc/adjtime`:

```
-9.600000 845082716 -0.250655
```

(¡Nótese que 9.6 segundos por día es aproximadamente cinco minutos al mes!)

3 Algunas consideraciones sobre `xntpd`

Su sistema actualmente dispone de dos relojes - el alimentado por la batería reloj de tiempo real (*real time clock*) que mantiene la hora cuando el sistema está apagado (también conocido como el reloj de la CMOS (*CMOS clock*, *Hardware clock* o *RTC*) y la hora del núcleo (*kernel time* o también conocido como *software clock* o *system clock*) que está basado en interrupciones de tiempo y que es inicializado con el reloj del CMOS en el momento del arranque. Ambos relojes pueden diferir en el tiempo que marcan, de hecho gradualmente difieren uno de otro a medida que pasa el tiempo, y también difieren de la "hora real".

Todas las referencias al "reloj" (*the clock*) en la documentación de `xntpd` se refieren a la hora del núcleo. Cuando ejecute `xntpd` o `timed` (o cualquier otro programa que ajusta la llamada del sistema de adjtimex), el núcleo de linux asume que la hora del núcleo es más precisa que el reloj del CMOS, y ajusta el reloj del CMOS cada 11 minutos (hasta que apague el sistema).

Esto significa que `clock` no sabrá más cuándo el reloj del CMOS fue ajustado por última vez, y no podrá utilizar el factor de corrección en `/etc/adjtime`. Puede usar `ntpdate` en su archivo de inicialización para poner en hora el reloj desde un servidor de tiempo en red, antes de iniciar `xntpd`. Si no dispone de una fuente de hora fiable de forma continua cuando arranque la máquina, puede ser algo complicado, ya que `xntpd` realmente no ha sido diseñado para ser usado en situaciones como esta.

`xntpd` incluye controladores para muchos radio relojes, y puede ser también puesto en hora llamando periódicamente al servicio telefónico NIST (asegúrese de calcular el efecto en su cuenta telefónica cuando configure el intervalo entre llamadas). Esto también sirve para obtener el factor de corrección al reloj del núcleo si se pierde el contacto con las otras fuentes de ajuste por un período largo de tiempo.

La mayoría de los radio-reloj cuestan $3-4K, pero puede hacer planes para tener una "caja negra" (actualmente un módem de 300 baudios) que esté ubicada entre su computador y una radio de onda corta sintonizada en la estación de tiempo CHU de Canadá (vea `ftp://ftp.udel.edu/pub.ntp.gadget.tar.Z`).

El receptor Heatkit WWV (el "reloj más exacto") está aún disponible (aunque no como kit), y cuesta alrededor de $4-500. La señal de los GPS también contiene información de la hora, y algunos receptores de GPS pueden ser conectados a un puerto serie. Esto podría constituir una solución de bajo costo en un futuro cercano.

En teoría, cualquiera puede escribir un programa para usar los servicios telefónicos de NIST, a fin de calcular la variación de tiempo entre el reloj del CMOS y el del núcleo automáticamente. No tengo noticia de ningún programa que haga esto por si sólo, pero la mayoría del código podría obtenerse de `xntpd`.

4 Anexo: El INSFLUG

El *INSFLUG* forma parte del grupo internacional *Linux Documentation Project*, encargándose de las traducciones al castellano de los Howtos (Comos), así como la producción de documentos originales en aquellos casos en los que no existe análogo en inglés.

En el **INSFLUG** se orienta preferentemente a la traducción de documentos breves, como los *COMOs* y *PUFs* (**P**reguntas de **U**so **F**recuente, las *FAQs*. `:)`), etc.

Diríjase a la sede del INSFLUG para más información al respecto.

En la sede del INSFLUG encontrará siempre las **últimas** versiones de las traducciones: `www.insflug.org`. Asegúrese de comprobar cuál es la última versión disponible en el Insflug antes de bajar un documento de un servidor réplica.

Se proporciona también una lista de los servidores réplica (*mirror*) del Insflug más cercanos a Vd., e información relativa a otros recursos en castellano.

Francisco José Montilla, `pacopepe@insflug.org`.