

El reloj

1. Conocer la hora

a. fecha

Para conocer la hora, utilice el comando **date**. Da la fecha actual, pero también calcula otras fechas en función o de la fecha actual o de cualquier fecha. **date** también modifica la fecha y hora del sistema.

```
$ date
sáb may 8 09:57:04 CEST 2021
```

Por defecto, la fecha visualizada es la fecha (y hora) local, configurada en función del huso horario. Para visualizar la hora UTC:

```
$ date -u
sáb may 8 07:57:42 UTC 2021
```

Se puede modificar el formato de la fecha a voluntad, tal y como se puede hacer con la función `C strftime`. En este caso, la sintaxis es:

fecha + "**formato**"

Veamos algunos ejemplos de formatos posibles:

| Formato | Resultado |
|---------|----------------------------------|
| %H | La hora tiene el formato 00..23. |
| %M | Minutos 00..59. |

| | |
|-----------|---------------------------------|
| %S | Segundos 00..60. |
| %T | Hora actual sobre 24 horas. |
| %r | Hora actual sobre 12 horas. |
| %Z | Huso horario. |
| %a | Día abreviado (lun, mar, etc.). |
| %A | Día completo. |
| %b | Mes abreviado. |
| %B | Mes completo. |
| %d | Día del mes. |
| %j | Día del año. |
| %m | Número del mes. |
| %U | Número de la semana 00..53. |
| %y | Dos últimas cifras del año. |
| %Y | Año completo. |

Para visualizar una fecha completa:

```
$ date +"Hoy es %A, %d %B %Y, son las %H horas, %M minutos y %S segundos"
```

```
Hoy es sábado, 08 mayo 2021, son las 10 horas, 04 minutos y 07 segundos
```

Puede modificar la base de cálculo con el parámetro `--date` seguido de una fecha o de un cálculo. Se aceptan las palabras clave today, yesterday, tomorrow, day(s), week(s), month(es), year(s), hour(s), minute(s), second(s) con + (añadido a la fecha) o - o ago (resta a la fecha especificada). Si no se especifica la fecha, será la fecha en curso.

Dentro de 10 días:

```
$ date --date "10 days"
```

```
mar may 18 10:04:26 CEST 2021
```

Mañana:

```
date --date "tomorrow"
```

Ayer:

```
date --date "yesterday"
```

Una semana después de la Navidad de 2020:

```
$ date --date "12/25/2017 23:59:00 + 1 week"
```

```
vie ene 1 23:59:00 CET 2021
```

b. hwclock

El comando **hwclock** permite interrogar directamente al reloj hardware RTC. El parámetro `--show` (por defecto) visualiza la fecha actual. Es diferente del tiempo del sistema que proviene de ntp o fecha.

```
# hwclock --show
2021-05-08 10:05:39.873892+02:00
```

No es posible formatear el resultado del comando.

2. Modificar el reloj físico

Sólo se puede modificar el reloj físico como root mediante los comandos **date** (reloj del sistema interno al núcleo) y **hwclock** (reloj físico). Observe que una modificación local de la fecha no funcionará de la forma deseada si el servicio NTP está configurado: la hora volverá a la hora proporcionada por un servidor NTP de forma sistemática.

a. Mediante date

Modifique la fecha y la hora con el parámetro `-s`:

```
# date -s "12/12/2016 12:41:00"
vie dic 12 12:41:00 CET 2016
# date
vie dic 12 12:41:03 CET 2016
```

b. Mediante hwclock

El comando **hwclock** modifica el reloj físico (RTC) o el reloj del sistema. Como el reloj material es independiente de la hora del sistema, los resultados pueden sorprender.

```
# hwclock --set --date "07/10/2014 14:00"
# date
domingo 29 enero 2017, 10:52:33 (UTC+0100)
# hwclock
2014-07-10 14:00:16.868788+1:00
```

Puede sincronizar la hora del sistema y la hora física en los dos sentidos. Para que se sincronice la hora física desde la hora del sistema:

```
# hwclock --systohc
```

Para realizar lo contrario:

```
# hwclock --hctosys
```

3. NTP

a. Fundamentos

NTP (*Network Time Protocol*) es un protocolo que permite sincronizar los relojes de los ordenadores mediante la red y, en particular, TCP/IP, o sea Internet. Como nuestros ordenadores utilizan relojes de cuarzo, a veces éstos se pueden adelantar o atrasar mucho dependiendo de la calidad de los componentes.

Existen numerosos ámbitos en los que no se puede admitir que un sistema no esté a la hora, en particular por razones de sincronización muy precisas.

Un servidor NTP emite la hora en formato UTC. El cliente recupera la hora y la adapta en función de su huso horario. El servidor no gestiona tampoco los cambios de hora.

Si el servidor NTP está al día, la hora es muy precisa. Se codifica en 64 bits:

- los primeros 32 bits dan el número de segundos desde el 1º de enero de 1900 a medianoche (de manera que el bug NTP tendrá lugar antes del bug Unix);
- los últimos 32 bits dan la precisión de los segundos.

El nuevo protocolo NTP4 da una precisión de los segundos en 64 bits, lo que evita un inoportuno error futuro, a través de los «ERA number» y «Era Offset».

Encontrará en la URL siguiente una lista de algunos servidores NTP españoles: <http://www.pool.ntp.org/es/zone/es>

b. Cliente NTP

El servicio `ntpd` permite sincronizar una máquina junto a un servidor de tiempo. Si no está instalado, habrá que hacerlo. Ejemplo en Ubuntu:

```
$ sudo apt-get install ntp ntpdate ntpstat
$ ps -ef|grep nt[p]
ntp      4302    1 0 10:23 ?    00:00:00 /usr/sbin/ntpd -p
/var/run/ntpd.pid -g -u 127:133
```

El archivo de configuración es `/etc/ntp.conf` o `/etc/ntp.conf`. En principio, este archivo ya contiene un determinado número de líneas que es preferible no tocar. Usted puede, incluso debe, añadir una línea que apunta al servidor de tiempo que ha elegido (por ejemplo es.pool.ntp.org):

```
server es.pool.ntp.org
```

Vuelva a iniciar el servicio. Su máquina debe ponerse en hora.

Es posible forzar una sincronización manual con el comando **ntpdate**. Este comando utiliza como parámetro un nombre de servidor ntp. El servicio NTP no debe estar activo.

```
# ntpdate es.pool.ntp.org
8 May 10:25:27 ntpdate[6092]: adjust time server 91.121.205.56 offset 0.000082 sec
```

Si no desea utilizar el servicio `ntpd`, puede colocar este comando en `crontab` todos los días o todas las horas.

El comando **ntpq** permite interrogar a los servidores NTP:

```
# ntpq -p
remote      refid      st t when poll reach  delay  offset jitter
=====
0.ubuntu.pool.n.POOL. 16 p - 64 0 0.000 0.000 0.000
```

```

1.ubuntu.pool.n .POOL.      16 p - 64 0 0.000 0.000 0.000
2.ubuntu.pool.n .POOL.      16 p - 64 0 0.000 0.000 0.000
3.ubuntu.pool.n .POOL.      16 p - 64 0 0.000 0.000 0.000
ntp.ubuntu.com .POOL.       16 p - 64 0 0.000 0.000 0.000
...
```

El comando **ntpstat** le proporciona una estadística en formato legible:

```

# ntpstat
synchronised to NTP server (163.172.10.91) at stratum 3
time correct to within 21 ms
polling server every 64 s
```

c. Deriva temporal

¿Qué ocurre cuando ocurre un problema de red, o cuando el reloj hardware del equipo no está bien ajustado ? La hora se ve desplazada lentamente, y después de algún tiempo, la variación entre la hora de sistema y la hora real se vuelve muy importante. Cualquier deriva de consideración es nefasta:

- ✧ Si la hora está adelantada o retrasada, la información de marcas de tiempo se convierte en falsa y las operaciones pueden ocurrir demasiado pronto o demasiado tarde (sistema de archivos, bases de datos, crontabs...).
- ✧ Si ajustamos la hora de una forma traumática, as operaciones que hubieran tenido lugar no se ejecutarán jamás, otras se ejecutarán dos veces...

Imagine los daños a los datos bancarios, o científicos... Si el servicio NTP detecta una deriva, pueden suceder dos cosas, con el objetivo de proteger el sistema:

- ✧ Si el delta es de menos de 128 milisegundos, NTP pasará de forma automática a la hora correcta.
- ✧ Si el delta es superior a 128 milisegundos, lo absorberá, **ajustando lentamente el reloj** del sistema por menos de 0,5 milisegundos hasta llegar a la hora correcta.
- ✧ Si se especifica el parámetro **-x**, el offset pasa de 128ms a 600s. Este parámetro

puede ser utilizado en un puesto de trabajo de oficina.

Si la deriva es superior a 1000 segundos, el servicio NTP no modificará la hora del sistema. Rehusará arrancar. Tendrá que ajustar la hora de su sistema de forma manual.

Una deriva de 600 segundos, en pasos de 0,5 ms, toma cerca de 14 días para corregir...

El parámetro **-g**, por defecto en muchas distribuciones, permite arrancar el servicio NTP con una deriva superior a 1000 segundos, permitiendo poner el sistema en hora de forma directa. Pero esto solo funcionará una sola vez; si una nueva deriva de más de 1000 segundos se produce, dejará de funcionar.

Para poner su sistema en hora si este parámetro no se ha especificado, detenga el servicio NTP y luego escriba el comando siguiente antes de reiniciarlo:

```
# ntpd -gq
```

4. timedatectl

La herramienta **timedatectl** está presente en muchas distribuciones. Permite leer y modificar la hora de manera práctica, a través de distintas fuentes, como el servicio NTP llamado **chrony**, aunque también usando el reloj interno de la máquina. Puede reemplazar los comandos **date** y **hwclock** para actualizar la hora del sistema.

```
# timedatectl
Local time: mar 2021-05-11 18:53:31 CEST
Universal time: mar 2021-05-11 16:53:31 UTC
RTC time: mar 2021-05-11 16:53:30
Time zone: Europe/Madrid (CEST, +0200)
System clock synchronized: yes
NTP service: active
RTC in local TZ: no
```

He aquí los comandos para cambiar la fecha o la hora del sistema:


```
# timedatectl set-time 10:26:30
# timedatectl set-time "2021-05-11 10:27:50"
```

También se pueden listar y modificar los usos horarios:

```
#timedatectl list-timezones | grep Europe/M
Europe/Madrid
Europe/Malta
Europe/Minsk
Europe/Monaco
Europe/Moscow
# timedatectl set-timezone Europe/Madrid
```

En algunas distribuciones basadas en systemd hay un servicio que se llama **systemd-timesyncd** que le permite gestionar la hora en lugar de NTP, o más bien usando sus comandos como **ntpdate** a intervalos regulares. En este caso, **timedatectl** puede interactuar con el servicio.

En la salida del primer ejemplo, la directiva **NTP service** contiene el valor **active**, lo que significa que hay un servicio NTP activado. Podría ser **ntpd** o **chrony**. Este último se conecta con timedatectl. Puede activar el soporte de ntp, de esta manera el comando se encargará de arrancar el servicio asociado:

```
# timedatectl set-ntp true
```

5. chrony

ntp no es el único en proponer un cliente de sincronización en Linux. Desde hace tiempo, **chrony** ha empezado a reemplazar ntp en la mayoría de las distribuciones. Se trata también de un cliente NTP, que se considera más simple y práctico. El cliente clásico NTP está más bien previsto para una máquina conectada 24 horas al día a la red, y encendida constantemente, chrony está adaptado tanto a los servidores como a los puestos de trabajo:

- ✓ Reemplaza ntp en las distribuciones mayores recientes (RHEL, SLES, Ubuntu, CentOS, Fedora, OpenSUSE).
- ✓ En el caso de que exista un desfase de tiempo importante, el ajuste gradual del tiempo se hará de una manera más rápida (algunos minutos) en chrony que en ntp.
- ✓ Reacciona con más rapidez, especialmente cuando el reloj interno del sistema no es fiable, como en el caso de las máquinas virtuales.
- ✓ Una vez que se haya hecho la sincronización inicial y el ajuste del desfase, el reloj es estable.
- ✓ La sincronización con los servidores NTP se realiza cuando la red está disponible, y no a intervalos regulares (uno de los grandes problemas del cliente NTP).

El servicio chrony es la mejor opción para una máquina que no esté constantemente conectada a la red o cuya actividad sea intermitente o que se ponga en modo suspendido.

La configuración de chrony se realiza en el archivo **etc/chrony.conf** .

```
# cat /etc/chrony.conf
pool 2.fedora.pool.ntp.org iburst
driftfile /var/lib/chrony/drift
makestep 1.0 3
rtcsync
keyfile /etc/chrony.keys
leapsectz right/UTC
logdir /var/log/chrony
```

Los valores interesantes son:

- ✓ **server**: un servidor NTP, con sus opciones.
- ✓ **pool**: define un grupo de servidores. La resolución del nombre de dominio devuelve varias direcciones IP.
- ✓ **rtcsync**: la hora del sistema se sincroniza cada 11 minutos con la hora RTC.
- ✓ **makestep**: permite controlar la velocidad del ajuste del desfase temporal forzando la actualización si el desfase es más importante que el indicado. En

este caso, si el desfase es más grande que 1 segundo en las tres primeras tentativas de ajuste, chrony pondrá la hora correcta directamente.

✓ **leapsectz**: cómo gestionar los segundos que se van intercalando en función del huso horario. El tiempo universal TU1, que es de hecho el tiempo solar medio, no es idéntico a UTC. Hay un desfase en el tiempo, que fuerza una incorporación de segundos. Desde 1972, se han añadido 37 segundos para mantener la sincronización.

Encontrará en Wikipedia más información sobre el tiempo solar y los desfases que se generan con el tiempo UTC en su página (en inglés): https://en.wikipedia.org/wiki/Solar_time

La dirección 2.fedora.pool.ntp.org le devuelve varias direcciones IP, por lo tanto, es un pool. Si ejecuta este comando varias veces, verá cómo las direcciones IP cambian:

```
# host 2.fedora.pool.ntp.org
2.fedora.pool.ntp.org has address 80.74.64.2
2.fedora.pool.ntp.org has address 51.15.175.180
2.fedora.pool.ntp.org has address 51.77.221.70
```

...

El comando **chronyc** se usa con parámetros o de manera interactiva. El parámetro **tracking** comprueba el estado de la sincronización:

```
$ chronyc tracking
Reference ID   : BD9FBE10 (2001:41d0:a:5a7::1)
Stratum       : 3
Ref time (UTC) : Tue May 11 18:58:20 2021
System time    : 0.000002539 seconds slow of NTP time
Last offset    : +0.003300448 seconds
RMS offset     : 0.003300448 seconds
Frequency      : 450.486 ppm slow
Residual freq  : +438.622 ppm
Skew           : 1000000.000 ppm
Root delay     : 0.029200219 seconds
Root dispersion : 5.882596970 seconds
```

Update interval : 2.0 seconds

Leap status : Normal

Para conocer las fuentes de sincronización:

```
$ chronyc sources
210 Number of sources = 4
MS Name/IP address      Stratum Poll Reach LastRx Last sample
=====
^* 185.83.169.27         1 10 377 467 +154us[ +133us] +/- 10ms
^+ erlenwald.stpeterscherts> 2 9 377 370 -170us[ -170us] +/- 17ms
^- time.rdg.uk.as44574.net 2 10 377 807 -466us[ -487us] +/- 46ms
^- ntp.uk.eria.one       2 6 377 10 +9160us[+9160us] +/- 21ms
```

Para obtener estadísticas de los grupos de sincronización:

```
# chronyc sourcestats
210 Number of sources = 4
Name/IP Address      NP NR Span Frequency Freq Skew Offset Std Dev
=====
185.83.169.27        31 12 154m +0.001 0.062 -55us 261us
erlenwald.stpeterscherts> 11 7 103m -0.059 0.262 +319us 270us
time.rdg.uk.as44574.net 37 20 174m -0.005 0.091 +27us 502us
ntp.uk.eria.one      6 3 324 +9.479 15.305 +12ms 414us
```

El siguiente comando le permitirá forzar la actualización de la hora:

```
# chronyc makestep
200 OK
```