# Servicios avanzados III: Sincronización de la hora mediante NTP.

**Autor: Enrique V. Bonet Esteban**

## *Introducción.*

De forma general, todo ordenador tiene un chip que le permite mantener la hora, tanto cuando esta funcionando como cuando esta apagado. Este chip suele ser puesto en hora de forma manual accediendo a los menús de configuración de la BIOS, o bien mediante la modificación de la hora con el sistema operativo que se utilice.

Esta sencilla operación suele ser valida para la mayoría de sistemas y ocasiones, pues, aunque el chip que mantiene la hora suele tener un error en su exactitud, basta con volver a sincronizar el reloj de forma manual al cabo de un cierto tiempo, generalmente meses.

Sin embargo, en ciertos ordenadores, como pueden ser los que controlan investigaciones donde la exactitud de la hora es fundamental para la experimentación (observaciones astronómicas por ejemplo), o en dispositivos de red que intercambian datos entre ellos, es necesario realizar un control mucho más exacto de la hora del ordenador.

Con el fin de proporcionar un mecanismo que permitiera sincronizar y mantener en hora el reloj de un ordenador se desarrolló el Network Time Protocol, el cual se encuentra descrito, en su versión actual (versión 4) en el RFC 2030.

El funcionamiento de NTP se basa en la existencia de una jerarquía dinámica de servidores de tiempo, de forma que todo servidor ocupa en un instante un nivel de la jerarquía. Los niveles de la jerarquía son Stratum-0, Stratum-1, ..., Stratum-16. De forma general, Stratum-0 son los relojes atómicos de cesio o los satélites de GPS. Los ordenadores conectados directamente a estos sistemas físicos forman el Stratum-1[1], los ordenadores que se sincronizan mediante los de Stratum-1 forman el nivel Stratum-2, etc. La jerarquía es dinámica pues un ordenador puede, en un momento dado, modificar su nivel por sincronizarse con un servidor de nivel superior o inferior al que utilizaba con anterioridad.

El mecanismo de sincronización es muy sencillo, cuando un ordenador arranca supone que su reloj se encuentra sincronizado correctamente de acuerdo a su chip en un nivel, que puede configurarse, y que suele ser Stratum-10. A partir del arranque, el ordenador empieza a intercambiar paquetes UDP con el puerto 123 de los ordenadores que tenga configurados como servidores de tiempo. El intercambio de paquetes con los servidores se produce por defecto a una frecuencia de un paquete cada 64 segundos ($2^6$), disminuyendo dicha frecuencia hasta 1024 segundos por defecto ($2^{10}$) a medida que se va produciendo la sincronización entre el cliente y los servidores. En este intercambio

[1] Un ordenador de Internet solo puede alcanzar el nivel Stratum-1, pues el Stratum-0 corresponde siempre a dispositivos físicos, no a ordenadores.

de información se eligen los mejores datos de los servidores[2] y se sincroniza el reloj del sistema con la hora proporcionada por los servidores de NTP, quedando al final nuestro sistema configurado como un posible servidor de nivel Stratum-(N+1), si Stratum-N es el servidor con el que al final se ha sincronizado nuestro ordenador.

El ordenador, una vez sincronizado, intercambiará cada 1024 segundos un paquete con los servidores conocidos, descartando los datos de los servidores en que la diferencia entre su reloj y el reloj de los servidores elegidos es superior a 128 milisegundos, excepto que durante más de 900 segundos se reciban todas las respuestas de los servidores con desviaciones superiores a 128 milisegundos, momento en el cual el ordenador considerará su reloj como no sincronizado y lo sincronizará nuevamente, utilizando los datos proporcionados por los servidores.

## *El servidor de NTP.*

Como hemos visto en la introducción, un servidor de NTP es tanto un cliente de NTP de servidores de Stratum superior al suyo, como un servidor de clientes de Stratum inferior al suyo[3].

El servidor de NTP es */usr/sbin/ntpd* y posee diversas opciones de inicialización, siendo las más importantes las especificadas en la siguiente tabla:

| **Opción** | **Descripción** |
|---|---|
| -c | Modifica el nombre del fichero de configuración que se utilizará. Por defecto el fichero es */etc/ntp.conf*. |
| -g | Por defecto ntpd termina su ejecución si la diferencia entre los servidores y la hora del sistema es superior a 1000 segundos. Con esta opción se elimina esta restricción. |
| -q | Termina la ejecución del programa una vez se ha actualizado por primera vez la hora. |
| -x | Actualiza la hora aunque el desfase sea inferior a los 128 milisegundos. |

El servidor posee cuatro modos básicos de funcionamiento que se establecen según el tipo de dirección IP a la que se requiere la sincronización. Estas se clasifican en:

- Direcciones clase *(s)*, que corresponden a un servidor remoto con dirección IP de clase A, B ó C.

- Direcciones clase *(b)*, que corresponden a direcciones broadcast de un interfaz local.

- Direcciones clase *(m)*, que corresponden a direcciones multicast, esto es, direcciones IP de clase D.

- Direcciones clase *(r)*, que indican un reloj de referencia. Se indican mediante una dirección IP no valida de tipo 127.127.t.u, donde t es un entero que indica el

[2] El algoritmo de selección de los datos, etc., queda fuera del ámbito de estos apuntes y no se explicará.
[3] Es posible configurar el servidor para que solo funcione como cliente y no como servidor.

tipo de reloj de referencia y n es un entero en el rango de 0 a 3 que indica distintos relojes dentro del mismo tipo de referencia[4].

Estos modos de funcionamiento se configuran, si no se especifica lo contrario, en el fichero */etc/ntp.conf*, el cual es un fichero de texto donde las líneas que comienzan por # son comentarios. Las opciones de funcionamiento se indican como:

```
server <dirección> [key <clave> | autokey] [burst] [iburst]
[version <versión>] [prefer] [minpoll <mínimo>] [maxpoll
<máximo>]

peer <dirección> [key <clave> | autokey] [version <versión>]
[prefer] [minpoll <mínimo>] [maxpoll <máximo>]

broadcast <dirección> [key <clave> | autokey] [version
<versión>] [prefer] [minpoll <mínimo>] [ttl <ttl>]

manycastclient <dirección> [key <clave> | autokey] [version
<versión>] [prefer] [minpoll <mínimo>] [maxpoll <máximo>] [ttl
<ttl>]
```

Donde *server* se aplica a las clases *(s)* y *(r)* e indica que el reloj del cliente se sincronizará siempre con el reloj del servidor, pero nunca al revés. Por su parte *peer* se aplica a la clase *(s)* e indica que el reloj local se sincronizará con el reloj remoto o viceversa en función de la situación en que se encuentre cada reloj. Por su parte, *broadcast* se aplica a las clases *(b)* y *(m)* e indica que se envíen mensajes a todas las direcciones broadcast indicadas. IANA ha asignado para este propósito la dirección broadcast 224.0.1.1 para su uso con NTP. Por último, la opción *manycastclient* se aplica solo a la clase *(m)* e indica mensajes multicast a la dirección que corresponde a los servidores multicast, que no deben corresponder a la 224.0.1.1 asignada por IANA a broadcast de NTP. De todas las opciones anteriores, la más utilizada y que corresponde a la estructura jerárquica que hemos indicado en la introducción es la opción *server*.

Una explicación de los parámetros de las opciones pueden encontrase en la tabla siguiente:

| **Parámetro** | **Descripción** |
|---|---|
| <dirección> | Dirección IP del servidor de NTP, puede indicarse como una dirección IP o como un nombre de ordenador. |
| key <clave> | Indica que todos los paquetes enviados y recibidos deben incluir un campo de autenticación encriptado usando un identificador de clave especificado por <clave>, que es un entero de 1 a 65534. Por defecto no se incluye campo encriptado[5]. |
| autokey | Similar al anterior pero se utiliza una clave autogenerada en lugar de una ya existente. |

[4] Una referencia de los valores de t y u, según los distintos relojes de referencia internos que pueden existir, se encuentra en la URL http://www.eecis.udel.edu/~mills/ntp/html/refclock.html.

[5] En estos apuntes no entraremos a discutir las opciones de autenticación, etc., pues tan solo aportan seguridad en la identificación de que los datos provienen de los servidores deseados.

| Parámetro | Descripción |
|---|---|
| burst | Si el servidor esta disponible se envían 8 paquetes en lugar de 1, dejando 16 segundos entre el primer y segundo paquete y 2 segundos entre el resto de paquetes. |
| iburst | Igual que el anterior pero solo si el servidor no esta disponible. |
| version <versión> | Especifica la versión de NTP a utilizar (1 a 4). Por defecto es 4. |
| prefer | Indica que si esta disponible este servidor será preferido para la sincronización. |
| minpoll <mínimo> | Tiempo mínimo entre dos mensajes NTP. Por defecto es 6 (64 segundos), pero puede reducirse hasta 4 (16 segundos). |
| maxpoll <máximo> | Tiempo máximo entre dos mensajes NTP. Por defecto es 10 (1024 segundos), pero puede aumentarse hasta 17 (36,4 horas). |
| ttl <ttl> | Indica el tiempo de vida de los paquetes broadcast o multicast. |

Para los diversos modos de funcionamiento existen opciones de configuración particulares de cada modo. Así, para el modo de funcionamiento *server*, cuando se utiliza para una clase *(r)* debe tener inmediatamente después la línea[6]:

```
fudge 127.127.t.u [stratum <entero>]
```

Donde el entero que acompaña a stratum sobrescribe el stratum por defecto del dispositivo de referencia, que suele ser 0.

Además, también se suele utilizar el comando *driftfile* para indicar un fichero donde se almacena la desviación de la frecuencia entre el reloj local y la obtenida por NTP. La sintaxis de dicho comando es:

```
driftfile <fichero>
```

Donde fichero indica el fichero donde se almacenará la desviación. Dicho fichero es actualizado cada hora y se utiliza, si existe, en el arranque del sistema como desviación de referencia entre la hora del reloj local y la exacta. En caso de que no exista se supone que dicha desviación es cero.

Por último, para el modo de funcionamiento *broadcast* y *manycastclient* se suele utilizar la opción de configuración *broadcastdelay*, cuya sintaxis es:

```
broadcastdelay <segundos>
```

Que indica el retraso de los paquetes entre el cliente y el servidor, pues en estos modos de funcionamiento puede no producirse en algunas ocasiones la calibración del retardo en la red. El valor por defecto si no se especifica es de 0.004 segundos.

Con lo visto hasta ahora, podemos configurar un sencillo cliente de NTP de forma que pueda sincronizar su hora con los servidores de NTP existentes en la red. Para ello, escribiremos el siguiente fichero de configuración[7]:

[6] La línea aquí indicada solo posee los parámetros necesarios para su uso con el reloj interno del sistema. Para otro tipo de relojes de referencia deberá consultarse una documentación más detallada de la opción.
[7] Una lista de los distintos servidores de NTP existentes, así como sus características, disponibilidad, etc., puede encontrarse en la URL http://www.ntp.org.

```
# Indicacion del reloj local en stratum-10
server 127.127.1.0
fudge 127.127.1.0 stratum 10
# Definicion de los servidores a utilizar
server 0.pool.ntp.org
server 1.pool.ntp.org
server 2.pool.ntp.org
# Fichero donde almacenar la diferencia del reloj local
driftfile /var/lib/ntp/drif
```

## Control de acceso.

La configuración anterior nos permite tener nuestro ordenador como cliente de NTP. Sin embargo, podemos desear que pueda suministrar la hora a otros clientes y convertirse en un servidor de NTP. En este caso, debemos controlar la información que suministra y las posibles acciones que pueden los clientes realizar sobre el servidor. Para ello, existen unos comandos que permiten controlar el acceso al servidor. Estos comandos son:

```
discard [average <valor>] [minimun <min>] [monitor <prob>]

restrict <dirección IP> [mask <máscara>] [bandera ...]
```

El comando *discard* configura los parámetros que protegen a un servidor del abuso de los clientes (siempre que el flag de *limited* sea configurado en el comando *restrict*). El parámetro *average* especifica el promedio mínimo de tiempo entre paquetes de un cliente (valor por defecto $2^5$), mientras que el parámetro *minimun* especifica el tiempo mínimo entre paquetes del cliente (valor por defecto $2^2$). Los paquetes que no cumplan esos tiempos mínimos son descartados y si autoriza en el comando *restrict*, se envía al cliente un paquete "beso de la muerte". Por último, el parámetro *monitor* especifica la probabilidad de descartar paquetes que no puedan ser monitorizados de forma correcta, su valor corresponde al número de paquetes que se admiten en un segundo.

El comando, *restrict* es el que permite indicar que pueden o no hacer los clientes. En primer lugar, la dirección IP corresponde a una dirección IP o red, especificándose mediante el formato *X.X.X.X*. La máscara, si existe, indica la máscara que se aplicará a la dirección para obtener la subred que indica la dirección IP. Por defecto, la máscara es *255.255.255.255*, lo que supone que la dirección IP corresponde a un ordenador y no a una subred. Existe una constante especial, denominada *default*, que permite especificar restricciones por defecto para todos los ordenadores que no se encuentren especificados por otras reglas *restrict*.

Por último, el campo *bandera* indica las restricciones que se aplican a los ordenadores indicados[8]. Por defecto, si no se especifica ninguna bandera, los clientes poseen un acceso completo al servidor. Las banderas y su descripción pueden verse en la tabla siguiente:

[8] Las banderas siempre indican restricciones, no existiendo ninguna bandera que permita ampliar o matizar la acción de otra bandera.

| Bandera | Descripción |
|---|---|
| ignore | Ignorar todos los paquetes provenientes de esta dirección IP. |
| kod | Si se deniega el acceso se envía un paquete del tipo “beso de la muerte”[9]. |
| limited | Deniega el servicio si los paquetes sobrepasan los limites establecidos con el comando discard. |
| lowpriotrap | Declara los mensajes atrapados de baja prioridad. |
| nomodify | Ignorar todos los paquetes de esa dirección IP que intenten modificar el servidor excepto las respuestas a las consultas realizadas, que obviamente modifican la hora del servidor. |
| noquery | Ignorar todos provenientes de esa dirección IP que soliciten consultas de información o configuración. |
| nopeer | Proporcionar servicio a esa IP solo si ya se estaba proporcionando servicio a la misma. |
| noserve | Ignorar los paquetes que no consultan o modifican el estado del servidor, esto es, consultas de sincronización de relojes. |
| notrap | No proveer el servicio de atrapar mensajes de control de paquetes de consulta del estado del servidor. |
| notrust | No utilizar esta IP como fuente de sincronización. |
| ntpport | Aplicar la restricción solo si el puerto origen del paquete es el de NTP (UDP 123). |
| non-ntport | Aplicar la restricción si el puerto origen del paquete no es el de NTP (UDP 123). |
| version | Ignorar el ordenador si no soporta la versión de NTP indicada. |

De forma general, todo servidor de NTP, aunque actúe solo como cliente, debe indicar opciones de control que permitan al sistema protegerse de posibles accesos exteriores que intenten modificar su comportamiento y no solo su hora.

Por ello, el ejemplo anterior de cliente de NTP debería ser modificado añadiendo las siguientes líneas:

```
# Restringimos las acciones de cualquier ordenador
restrict default nomodify notrap noquery
# Permitimos cualquier tipo de accion desde loopback
restrict 127.0.0.1
# Indicacion del reloj local en stratum-10
server 127.127.1.0
fudge 127.127.1.0 stratum 10
# Definicion de los servidores a utilizar
server 0.pool.ntp.org
server 1.pool.ntp.org
server 2.pool.ntp.org
# Fichero donde almacenar la diferencia del reloj local
driftfile /var/lib/ntp/drif
```

[9] Por defecto un servidor no responde a un cliente al que no desea dar servicio. Sin embargo, si el servidor envía como respuesta un paquete de tipo “beso de la muerte” el cliente, al recibir el mismo, interpreta que el servidor no desea atenderle y que debe marcarlo como que deniega el servicio, si es el primer paquete enviado, o bien como que se ha producido una limitación del número de clientes en el servidor si no es el primer paquete.

Donde podemos ver que hemos añadido una opción de restricción que no permite a ningún ordenador realizar consultas al servidor de NTP, por lo que actúa tan solo como cliente, ni modificar su estado, junto con una línea que permite modificar cualquier valor del servidor desde su interfaz de loopback, esto es, de forma local.

Si una vez configurados nuestro ordenador como cliente de NTP deseamos que actúe como servidor de NTP de Stratum-(N+1), podemos modificar el fichero anterior de la siguiente forma:

```
# Permitimos solo la consulta desde cualquier ordenador
restrict default nomodify notrap
# Permitimos cualquier tipo de accion desde loopback
restrict 127.0.0.1
# Indicacion del reloj local en stratum-10
server 127.127.1.0
fudge 127.127.1.0 stratum 10
# Definicion de los servidores a utilizar
server 0.pool.ntp.org
server 1.pool.ntp.org
server 2.pool.ntp.org
# Fichero donde almacenar la diferencia del reloj local
driftfile /var/lib/ntp/drif
```

Permitiendo la consulta de cualquier ordenador de Internet a nuestro servidor de Stratum-(N+1). Si deseamos, por ejemplo, que solo el ordenador *147.156.222.65* pueda utilizar nuestro ordenador como servidor de NTP, la modificación que deberíamos realizar es la siguiente:

```
# Restringimos las acciones de cualquier ordenador
restrict default nomodify notrap noquery
# Permitimos ser servidores del ordenador 147.156.222.65
restrict 147.156.222.65 nomodify notrap
# Permitimos cualquier tipo de accion desde loopback
restrict 127.0.0.1
# Indicacion del reloj local en stratum-10
server 127.127.1.0
fudge 127.127.1.0 stratum 10
# Definicion de los servidores a utilizar
server 0.pool.ntp.org
server 1.pool.ntp.org
server 2.pool.ntp.org
# Fichero donde almacenar la diferencia del reloj local
driftfile /var/lib/ntp/drif
```

## *Los comandos ntpstat y ntpq.*

Los comandos *ntpstat* y *ntpq* permiten conocer el estado de nuestro cliente de NTP.

El primero de ellos, */usr/bin/ntpstat* nos indica el estado de sincronización de nuestro cliente, el servidor con el que está sincronizado, el stratum de la sincronización

y el tiempo de desviación entre nuestro reloj y el del servidor, así como el intervalo de tiempo entre una consulta y la siguiente. Un ejemplo de su ejecución es el siguiente[10]:

```
#ntpstat
synchronised to NTP server (129.215.160.240) at stratum 3
   time correct to within 78 ms
   polling server every 1024 s
```

Por su parte, el comando */usr/sbin/ntpq* permite consultar el estado del servidor de NTP. El comando posee muchas opciones, aunque la que más nos interesa es la que nos permite mostrar la información que posee el cliente de NTP de los servidores de NTP a los que consulta. Esta información se puede obtener ejecutando en la línea de comandos el comando:

```
/usr/sbin/ntpq –p
```

O bien, si hemos ejecutado el comando anterior sin la opción *-p* y hemos entrado en el modo interactivo, escribiendo:

```
>peers
```

En ambos casos, obtenemos la siguiente respuesta:

```
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
+ns2.puck.ch     194.42.48.120    3 u  394 1024  377   33.018   -0.087   0.382
+gaia.ailab.ch   129.132.2.21     3 u  391 1024  377   32.398   -0.815   0.126
*linnaeus.inf.ed 129.215.64.241   2 u  432 1024  377   44.179    0.439   0.370
```

Donde el primer carácter de cada línea indica el estado del servidor según la siguiente tabla:

| **Carácter** | **Descripción** |
|---|---|
| Espacio | El servidor ha sido descartado por no poder consultarse o ser local. |
| x | El servidor ha sido descartado por problemas de distancia de sincronización. |
| - | El servidor ha sido descartado por ofrecer respuestas incorrectas. |
| + | El servidor es utilizado en el algoritmo de cálculo. |
| # | El servidor es candidato a entrar en el algoritmo de calculo si alguno de los existentes no se encuentra disponible. |
| * | El servidor es el elegido actualmente como servidor de referencia. |

Después de ese primer carácter viene la IP del servidor (*remote*), la IP del servidor de referencia o tipo de dispositivo físico con el que se sincroniza el servidor (*refid*), el stratum de ese servidor (*st*), su estado (*t*), cuantos segundos han transcurrido desde la última consulta al mismo (*when*), cuando tiempo transcurrirá entre consultas a ese servidor (*poll*), el resultado de las últimas 8 consultas en octal con la última consulta indicada en el bit más bajo (*reach*)[11], el tiempo que tarda, en milisegundos, en llegar la

[10] En la contestación podemos ver que nuestro servidor es de stratum-2, pues está sincronizado a un servidor de stratum-1. Si en un momento dado nuestro servidor se sincronizará con un servidor de distinto stratum su stratum cambiaría.

[11] Si una consulta tiene éxito el bit correspondiente tiene valor 1. Así, 375 en octal se traduce a 11111101 que indicaría que la penúltima consulta no fue contestada mientras que el resto de las últimas 8 si que fueron contestadas.

respuesta desde el servidor a nuestro cliente (*delay*), la diferencia en milisegundos entre el reloj de ese servidor y el nuestro (*offset*) y la desviación existente, también en milisegundos, entre las distintas respuestas obtenidas de ese servidor (*jitter*)[12].

[12] La desviación mide la calidad de las respuestas de ese servidor, de forma que una desviación pequeña indica que las respuestas del servidor son muy aproximadas entre ellas.