



Part 38

-

Getting The Right Time

NTP Client

One of disadvantages designing so cheap computer is that you have to give up some features which are too expensive. One of these features is RTC - real-time clock. RTC chip has its own battery to store actual time even when device isn't plugged in.

Problem with storing actual time is partially solved by fake-hwclock package which writes actual time to text file every hour and at system reboot. At start, if no better source of time is present, time is set to one stored in that text file. Since RPi is disposing a RJ-45 port, we can use it as very precise time source within NTP. First we have to install package called `ntp`.

```
sudo apt-get -y install ntp
```

Notice: the newer version of Raspbian have preinstalled timesyncd service. It can be used for the same purpose, but you can not have running both of them at the same time. To fix this you can stop and disable timesyncd service using these commands.

```
systemctl stop systemd-timesyncd
systemctl disable systemd-timesyncd
/etc/init.d/ntp stop
/etc/init.d/ntp start
```

That's all, RPi is now synchronizing its time to NTP servers. By default it uses NTP servers which are generally too far from you. It has bad influence to time accuracy. So first go to page pool.ntp.org and find a location as near as possible. For me it is Belgium. To set NTP servers, open `ntp` configuration file.

```
nano /etc/ntp.conf
```

And find lines starting with "server". Replace that lines for lines with servers from pool.ntp.org. In my case configuration looks like:

```
#/etc/ntp.conf, configuration for ntpd

driftfile /var/lib/ntp/ntp.drift
statsdir /var/log/ntpstats/

statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable

# You do need to talk to an NTP server or two (or three).
#server ntp.your-provider.example

# pool.ntp.org maps to more than 300 low-stratum NTP servers.
# Your server will pick a different set every time it starts up.
# *** Please consider joining the pool! ***
# *** ***

server 0.be.pool.ntp.org iburst
server 1.be.pool.ntp.org iburst
server 2.be.pool.ntp.org iburst
server 3.be.pool.ntp.org iburst

# By default, exchange time with everybody, but don't allow configuration.
# See /usr/share/doc/ntp-doc/html/accept.html for details.
```

```

restrict -4 default kod notrap nomodify nopeer noquery
restrict -6 default kod notrap nomodify nopeer noquery

# Local users may interrogate the ntp server more closely.
restrict 127.0.0.1
restrict ::1

# Clients from this (example!) subnet have unlimited access,
# but only if cryptographically authenticated
#restrict 192.168.123.0 mask 255.255.255.0 notrust

# If you want to provide time to your local subnet, change the next line.
# (Again, the address is an example only.)
#broadcast 192.168.123.255

```

Now you can restart ntp service.

```
/etc/init.d/ntp restart
```

Here is command to check if time is synchronizing properly. To list NTP servers with which RPi is synchronizing:

```
ntpq -pn
```

You will get some output like this

| remote | refid | st | t | when | poll | reach | delay | offset | jitter |
|-----------------|----------------|----|---|------|------|-------|--------|--------|--------|
| *162.159.200.1 | 10.78.8.101 | 3 | u | - | 64 | 1 | 17.176 | -0.466 | 1.727 |
| 109.68.160.220 | 10.0.0.241 | 2 | u | 1 | 64 | 1 | 19.978 | -1.037 | 2.781 |
| 91.121.216.238 | 193.190.230.66 | 2 | u | - | 64 | 1 | 24.746 | 2.173 | 1.479 |
| 193.190.253.212 | 193.79.237.14 | 2 | u | 1 | 64 | 1 | 16.927 | -0.846 | 0.762 |

First letter have the most significant value of the table. * means actual time source, + means something like backup time source, - is ignored, but can become backup (+) at any time.

"remote" shows IP addresses of NTP servers.

"refid" shows source of time for the NTP server.

"st" shows so called stratum. Stratum is number from 0 to 15 and shows number of hops from reference time (GPS clock, atomic clock...). That device has stratum 0. Maybe you can also find stratum 16 which means untrusted time source.

"t" with "u" means that NTP server's address is unicast.

"when" indicates number of seconds passed since last response.

"poll" means interval between time requests, in seconds.

"reach" indicates success/failure to reach source.

"delay" indicates the roundtrip time, in milliseconds, to receive a reply.

"offset" is the difference between ntp server and your system clock in milliseconds.

"jitter" shows the difference, in milliseconds, between two samples.

Now you can check your time... I like to do everything in right way - so this command will print actual time in ISO 8601 format (representation of dates and times standard)

```
date +"%FT%T%Z"
```


Using systemd as NTP Client

Debian is by default using 0.debian.pool.ntp.org ... 3.debian.pool.ntp.org to find an NTP server.

Configuration is stored in /etc/systemd/timesyncd.conf configuration file.

```
$ cat /etc/systemd/timesyncd.conf
# This file is part of systemd.
#
# systemd is free software; you can redistribute it and/or modify it
# under the terms of the GNU Lesser General Public License as published by
# the Free Software Foundation; either version 2.1 of the License, or
# (at your option) any later version.
#
# Entries in this file show the compile time defaults.
# You can change settings by editing this file.
# Defaults can be restored by simply deleting this file.
#
# See timesyncd.conf(5) for details.
[Time]
#NTP=
#FallbackNTP=0.debian.pool.ntp.org 1.debian.pool.ntp.org
#2.debian.pool.ntp.org 3.debian.pool.ntp.org
#RootDistanceMaxSec=5
#PollIntervalMinSec=32
#PollIntervalMaxSec=2048
```

Modify this configuration file to use specific NTP servers.

```
$ sudo nano /etc/systemd/timesyncd.conf

[Time]
NTP=time.google.com
FallbackNTP=0.debian.pool.ntp.org 1.debian.pool.ntp.org
#2.debian.pool.ntp.org 3.debian.pool.ntp.org
#RootDistanceMaxSec=5
#PollIntervalMinSec=32
#PollIntervalMaxSec=2048
```

Disable NTP service for a moment to reload configuration.

```
$ sudo systemctl stop systemd-timesyncd
$ sudo timedatectl set-ntp false
```

Wait a second

```
$ sudo timedatectl set-ntp true
$ sudo systemctl start systemd-timesyncd
```

Inspect systemd service responsible for time synchronization.

```
$ systemctl status systemd-timesyncd.service
● systemd-timesyncd.service - Network Time Synchronization
   Loaded: loaded (/lib/systemd/system/systemd-timesyncd.service; enabled;
   vendor preset: enabled)
```

```
Drop-In: /usr/lib/systemd/system/systemd-timesyncd.service.d
          └─disable-with-time-daemon.conf
Active: active (running) since Mon 2019-06-03 22:05:08 GMT; 2min 1s ago
        Docs: man:systemd-timesyncd.service(8)
Main PID: 2299 (systemd-timesyn)
Status: "Synchronized to time server for the first time 216.239.35.0:123
(time.google.com)."
Tasks: 2 (limit: 394)
Memory: 1.3M
CGroup: /system.slice/systemd-timesyncd.service
         └─2299 /lib/systemd/systemd-timesyncd
```

Display detailed service status.

```
$ timedatectl timesync-status
      Server: 216.239.35.0 (time.google.com)
Poll interval: 1min 4s (min: 32s; max 34min 8s)
      Leap: normal
      Version: 4
      Stratum: 1
      Reference: GOOG
      Precision: 1us (-20)
Root distance: 213us (max: 5s)
      Offset: -14.998ms
      Delay: 35.315ms
      Jitter: 38.186ms
Packet count: 3
      Frequency: +382.821ppm
```

Display detailed service status in machine readable form.

```
$ timedatectl show-timesync
SystemNTPServers=time.google.com
FallbackNTPServers=0.debian.pool.ntp.org 1.debian.pool.ntp.org
2.debian.pool.ntp.org 3.debian.pool.ntp.org
ServerName=time.google.com
ServerAddress=216.239.35.0
RootDistanceMaxUsec=5s
PollIntervalMinUsec=32s
PollIntervalMaxUsec=34min 8s
PollIntervalUsec=1min 4s
NTPMessage={ Leap=0, Version=4, Mode=4, Stratum=1, Precision=-20,
RootDelay=0, RootDispersion=213us, Reference=GOOG, OriginateTimestamp=Mon
2019-06-03 22:06:13 GMT, ReceiveTimestamp=Mon 2019-06-03 22:06:13 GMT,
TransmitTimestamp=Mon 2019-06-03 22:06:13 GMT, DestinationTimestamp=Mon
2019-06-03 22:06:13 GMT, Ignored=no PacketCount=3, Jitter=38.186ms }
Frequency=25088557
```

Display specific information.

```
$ timedatectl show-timesync --property ServerName --value
time.google.com
```

NTP Server

Now when RPi has synchronized its time, you can set it to pass this time information to devices in your local network. You only have to open configuration file of ntp daemon.

```
nano /etc/ntp.conf
```

And add a string that describes the hosts which requests will be answered.

```
restrict 192.168.1.0 mask 255.255.255.0
```

Note: my LAN uses 192.168.1.0/24 hence the IP range above

Also add the next 2 lines. They will enable sending of broadcasts and multicasts containing time information for devices which accept them (Cisco, Juniper...)). Do not change multicast address 224.0.1.1 since this addresses is assigned to NTP service by IANA and some network devices join this multicast group automatically.

```
broadcast 192.168.1.255  
broadcast 224.0.1.1
```

Now close the configuration file (CTRL+X...) and save changes (...press "y" and Enter). Last step is to restart ntp daemon with:

```
/etc/init.d/ntp restart
```

From now you can configure machines in local network to synchronize its time to RPi.

SNTP in Windows XP

First versions of Windows to Windows 8 still don't have support for network time protocol. They use Simple network time protocol instead which have a limited accuracy. Accuracy is matter of up to 4 seconds which is much more than NTP's few milliseconds :) SNTP in Windows synchronize the time once a week, NTP daemon in Debian synchronizes its time approximately every 512 seconds by default...

timedatectl

The last Raspbian version (lite or desktop) includes `timedatectl` by default. It's a tool to manage the date and time on the Raspberry Pi. The first command is how to check the current status

```
timedatectl status
```

This should give you something like this:

```
root@MeRasPi3B_WeatherStation:~# timedatectl status
    Local time: Fri 2019-11-01 19:26:48 CET
    Universal time: Fri 2019-11-01 18:26:48 UTC
        RTC time: n/a
        Time zone: Europe/Brussels (CET, +0100)
    NTP enabled: no
NTP synchronized: yes
    RTC in local TZ: no
        DST active: no
Last DST change: DST ended at
                  Sun 2019-10-27 02:59:59 CEST
                  Sun 2019-10-27 02:00:00 CET
Next DST change: DST begins (the clock jumps one hour forward) at
                  Sun 2020-03-29 01:59:59 CET
                  Sun 2020-03-29 03:00:00 CEST
root@MeRasPi3B_WeatherStation:~#
```

So you'll get:

- The local time
- The universal time (same thing by default)
- The RTC time if configured (module not included on Raspberry Pi)
- The current time zone (GMT by default)
- The current network time synchronization status

As you can see, my Raspberry Pi is already time synchronized by default (except the time zone)

Before going further in the `timedatectl` configuration, I want to show you some useful commands you can use directly

- *List time zones*

If you need to change the default time zone, you first need to know all available values
To do this, use this command:

```
timedatectl list-timezones
```

As the list is big, you can filter it with the `grep` command

```
timedatectl list-timezones | grep America
timedatectl list-timezones | grep Sydney
```

Note your local time zone and use it with the next command

- *Set time zone*

To set the current time zone, use this command:

```
sudo timedatectl set-timezone <time zone>
```

For example:

```
sudo timedatectl set-timezone America/New_York  
sudo timedatectl set-timezone Europe/Paris  
sudo timedatectl set-timezone Australia/Sydney
```

Use again the `timedatectl status` to check that the current time is correct
You can also change the time zone in raspi-config > Localization options > Change time zone

- *Set the time manually*

It's not really in the state of mind of this post, but you can set the time manually with `timedatectl`

Here is how:

```
sudo timedatectl set-time 'A:M:J HH:mm:ss'  
sudo timedatectl set-time 'A:M:J'  
sudo timedatectl set-time 'HH:mm:ss'
```

For example:

```
sudo timedatectl set-time '12:00:00'
```

But to do this, you need to disable the time synchronization (see next paragraph)

- *Enable or disable the time synchronization*

If you want to disable or enable the time synchronization, use these commands:

```
sudo timedatectl set-ntp false  
sudo timedatectl set-ntp true
```

Note: you may need to reboot the Raspberry Pi to apply this change (see comments)
That's it, you know the basic `timedatectl` commands