

Setup on Windows


[Home Page](#)[Up](#)[Setup on Windows](#)[Using NTP](#)[Windows LAN tips](#)[Performance](#)[Events](#)[Cable modem notes](#)[Monitoring with MRTG](#)[GPS 18 + FreeBSD](#)[GPS 18 + Windows](#)[GPS 18x firmware](#)[NTP 4.2.4 vs. 4.2.5](#)[NTP 4.2.7p241](#)[Rapco 1804M notes](#)[Raspberry Pi](#)[RPI - cross-compile](#)[Sure GPS board](#)[Timestamp issues](#)[TSC Interpolation](#)[Vista & Windows-7/8](#)[Wi-Fi warning](#)[Windows after reboot](#)[Win-8 with Internet](#)[Development versions](#)

Setting up NTP on Windows

It's very helpful that [Meinberg](#) have provided a port of the highly-respected and high-accurate [NTP](#) software for Windows users - my thanks to them. This Web page provides a quick guide to installing NTP on Windows XP, Vista, or Windows-7/8, using the standard out-of-the-box settings to sync your PC to the Internet, and then checking that you have a working installation. [These notes](#) show why you would want to use NTP as opposed to other timekeeping software. Some programs such as the excellent [Plane Plotter](#) even *require* that NTP be installed before you are allowed to use some of the more advanced features.

If you have more than one PC, you should install NTP as described below on each PC. If you have more than a few PCs you may want to create one local time server (locked to the Internet) for your network and sync the other PCs to that local server, by adding one line to the configuration file. More information is [here](#). This is easy, because the software will work as both a client to a remote NTP server, *and* as a server to which other NTP clients can connect. For a rather low extra cost (US \$35, £25) you can [lock that local time server to GPS](#), making it far more precise than one locked to Internet sources. You might like to use something like a [Raspberry Pi](#) as a low-cost, stand-alone, precision time server.

Installing

Visit the [Meinberg Web site](#) (opens in a new tab), and look for the section entitled: **NTP for Windows XP and newer, with IPv6 support**. Look for the download icon  and a file named ntp...setup.exe. Download the setup file and save it to your hard disk.

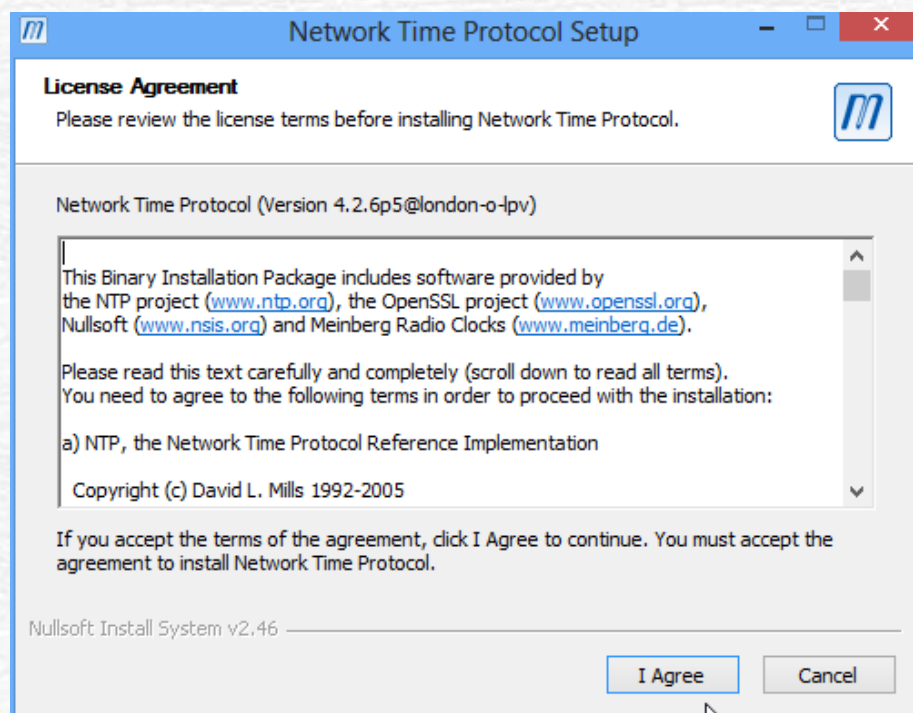
Where to download software installation files

I normally make a directory named C:\Install\ on new PCs, and it's where I put all my software which I've downloaded. So in this case, I create a new directory named C:\Install\NTP\, and download the ...setup.exe file directly to that directory.

As of July 2014, the current Meinberg version was named: ntp-4.2.6p5@spider-man-2-win32-setup.exe (3.61 MB)

I suggest right-clicking the downloaded file, select Properties, and clicking the Unblock button if it is present. This will remove one warning message before running the install procedure. Then run the setup.exe file by double-clicking. On Vista or Windows-7 you may need to allow the file to run by answering the UAC (User Authorisation Control) question, however, you *don't* have to disable UAC either to install or to run NTP. I have had one report that on a Windows-XP/64 installation, using right-click, Run as Administrator was required, otherwise

the account created for NTP had insufficient privileges. The screenshots below are from an earlier Windows-8 installation.



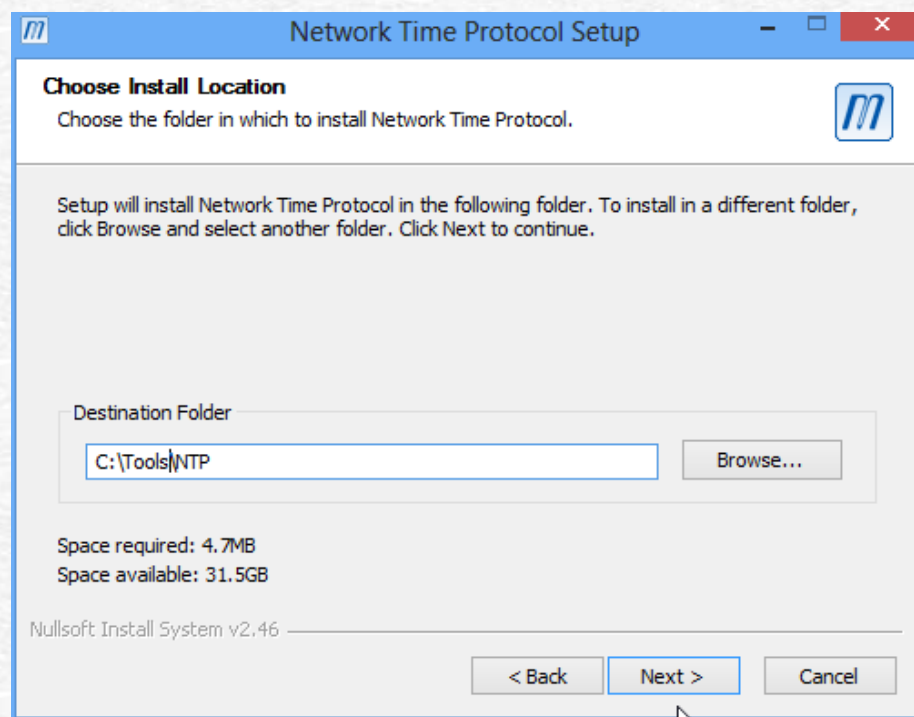
Assuming you agree to the terms, press I Agree. Select a location for the files - you have a choice to make here.

Where to install NTP?

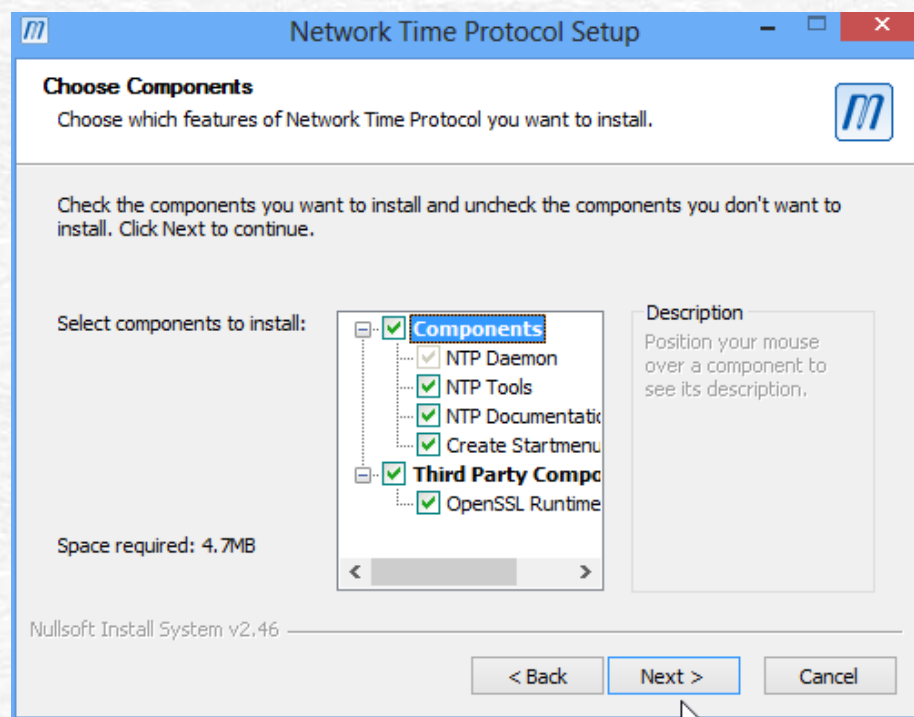
In a single-user PC, in a benign environment, I recommend creating a directory called `C:\Tools\` if you don't already have one, and putting the NTP software into `C:\Tools\NTP\`. I suggest this as the installation includes user-editable files, and `C:\Program Files\` isn't an appropriate location for such data. This is because of the directory virtualisation which Windows Vista and Windows-7/8 perform, and you end up editing a file, but it's not the file which NTP will read.

When you create the `C:\Tools\` directory, give all users full control, by right-clicking the directory in Windows Explorer, selecting Properties, Security tab, Edit, Add. In the object names box, type "everyone", OK, ensure that "Everyone" is selected, and in the Permissions box, click on Full Control, OK.

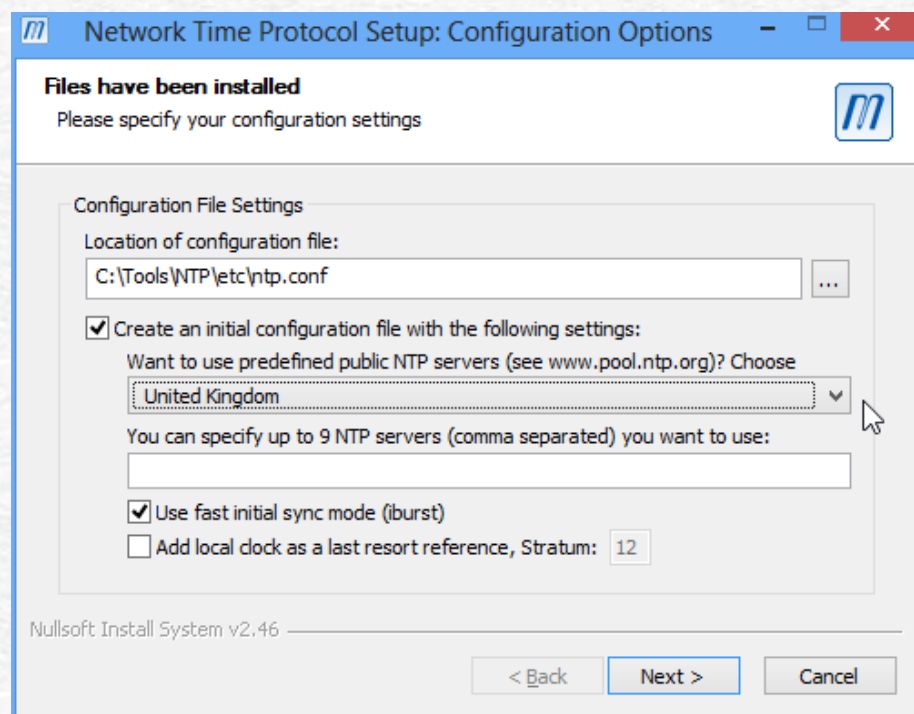
In a multi-user PC or one which is in a not so friendly environment, leave the default install destination folder as `C:\Program Files\NTP\` (or whatever is suggested), but be aware that you may need to have Administrator rights when editing the configuration file. If you don't want to edit the configuration file, and most users will not, it's not a great inconvenience.



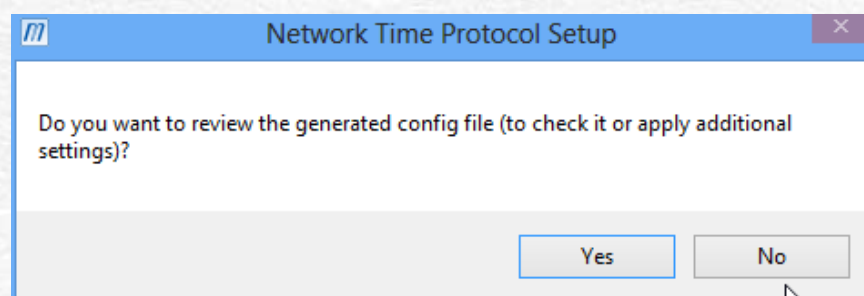
Choose your directory and press Next to continue.



Leave all the components selected, and press Next to continue. NTP can create an initial configuration for you using servers from the [NTP Pool](#) based on your geographic location. You *should* allow it to do this, as it saves you the task of choosing which servers to use. Ensure the box "Create an initial..." is checked, and select your nearest country or region from the drop-down list. In the example below, I have selected United Kingdom. NTP will then choose servers in your region to try and provide the nearest connection for best performance. Leave the other settings as the installer suggests.

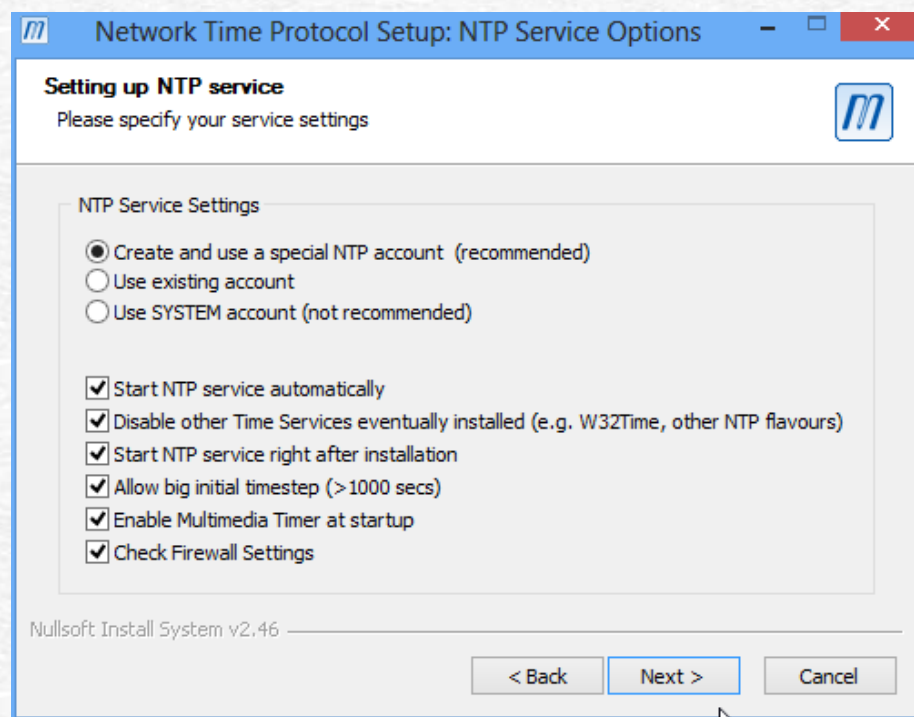


Once you have selected your region, press Next. Most users will not need to edit the file which the installer creates, so press No to continue.

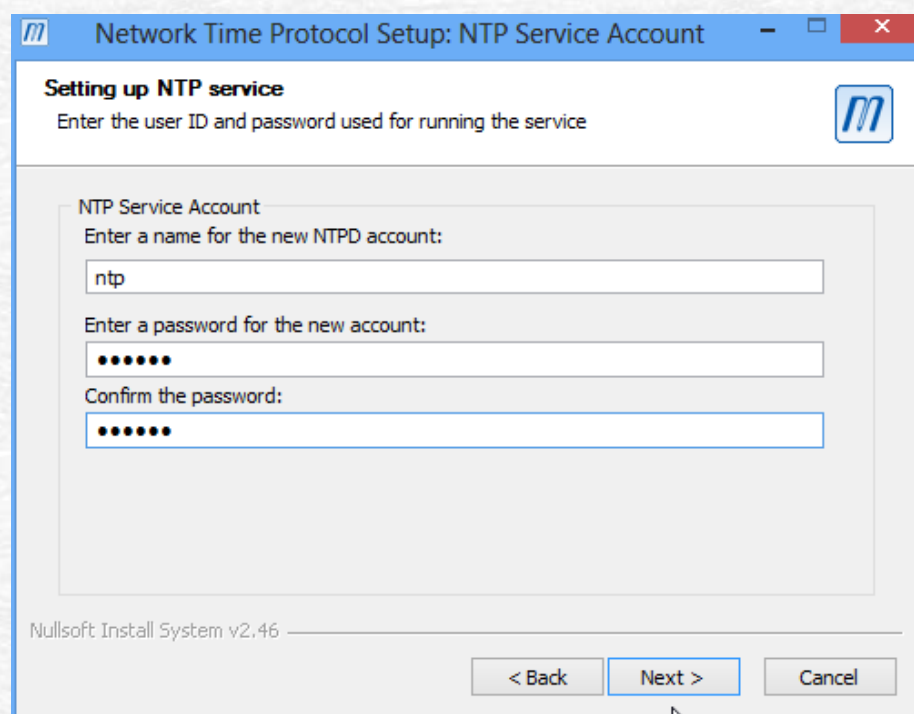


NTP can create an account to run the software. NTP will run even while you are not logged into the computer ensuring even better timekeeping. The next dialog recommends that the installer will create an account for you, and that is the appropriate setting. It is strongly recommended *not* to use the SYSTEM account. Leave the other options as suggested.

One user reported problems as he had unchecked the "Disable other Time Services..." box below, and this led to Windows and NTP fighting over the time setting, and NTP being unable to do its job. Unfortunately, the initial install had saved its settings in the file install.ini, so that when a re-install was attempted the same incorrect setting persisted. Editing the install.ini file restored the correct defaults and allowed a successful installation. Do check carefully if you are installing for a second time.



Press Next and define a password for the account. *Be sure to make a note of this password* in case you need it in the future or for a re-install, but as the account has limited rights there is normally no need for it to be an ultra-secure password. Should you forget that password, you would need to remove the `ntp` account in the unlikely event of a re-install. One person got an [Error 2245](#) from using too simple a password, or re-using the same password, but I've had no other reports.

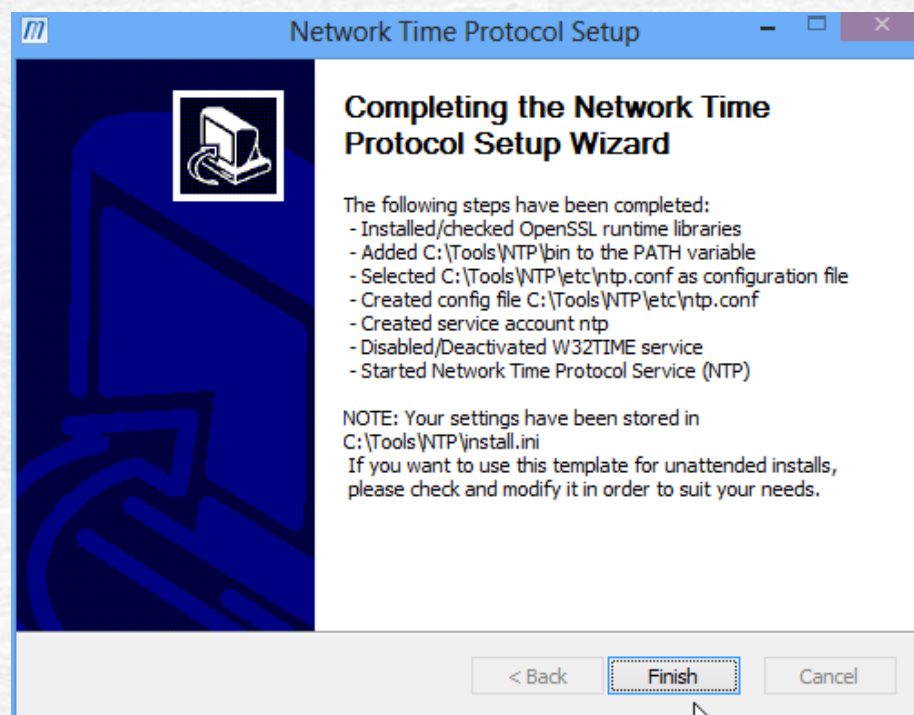


Once you press Next, the installation completes, and the NTP service is started.

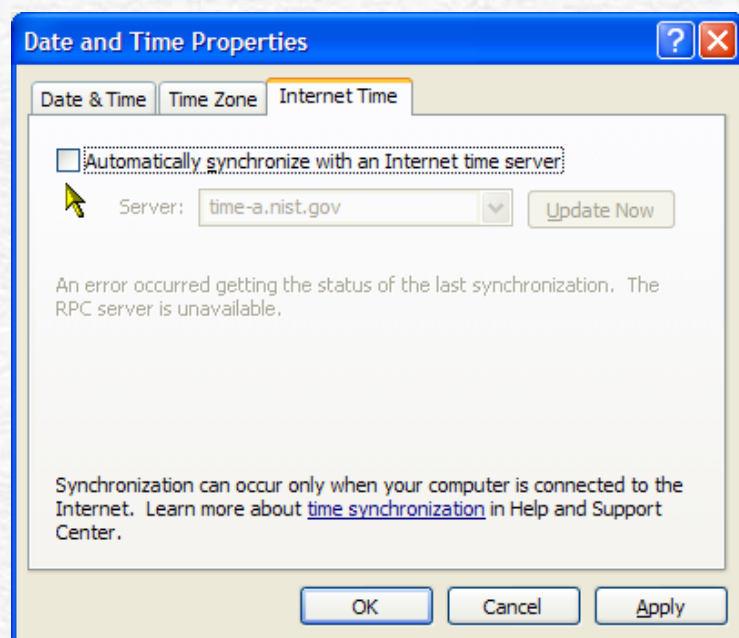
You may need to allow programs through the Windows Firewall

Like any programs which access the Internet, NTP may trigger your firewall software into asking you if it's OK for NTPD to access the Internet. Be sure to allow `ntpd.exe` both incoming and outgoing access (called "Act as a server" in Zone Alarm). There is also a support program (`ntpq.exe`) you can use to query how well NTP is working

either on your own system or any system you have access to in you locally or on the Internet, and that may also need to be allowed through your firewall. [How to add a program to the firewall for Windows-7](#). The programs will be found in `C:\Tools\NTP\bin\` if you have been using the suggested paths. Allow `ntpd.exe` to accept incoming UDP requests through the firewall if you intend to set up a local NTP network. If you haven't disabled the hiding of file extensions the programs may show simply as "ntpd" and "ntpq". I do recommend that you [allow Windows Explorer to show file extensions](#) so that you will be less confused.



Although it should not be necessary, you may want to review the Control Panel, Date and Time dialog, the check-box suggesting that Windows automatically synchronise with an Internet server. There is no need for this setting as NTP does a *much* better job, indeed - this box *must* be unchecked for NTP to work correctly.



You can now repeat this installation for each of the PCs on your network.

Routers

No special configuration should be needed for the majority of routers, as support for NTP (UDP packets on port 123) is already built-in.

Using with a virtual PC

VMware

VMware publish their own [recommendations for using NTP](#) in a virtual PC environment. These recommendations have changed with the various software releases, so it's best to check the [VMware Web site](#) for the current ideas.

Microsoft Hyper-V

With the Microsoft Hyper-V 2012 software, some suggestions from Steve Walker (who needs good timekeeping for the excellent [Plane Plotter software](#)) are:

- Host PC configuration:
 - NTP: installed and working
 - OS: Microsoft Windows Server 2012 Standard
 - Virtual PC: Hyper-V 2012, with the Time Sync option enabled for the client
 - See [this screen-shot](#), where the setting is shown in its disabled state, you want to be sure it's checked.
- Virtual PC configuration:
 - OS: Microsoft Windows XP SP3
 - Application software: Plane Plotter 6.3.6

Steve Walker comments: I made sure the Hyper-V Time Synchronisation Service is started, as this will then allow the client to sync its time with the host.

Domain Time II

If the timekeeping from Hyper-V Time Synchronisation Service isn't good enough for you, you may be interested in Domain Time II noted by Larry Ellis - *he writes*:

I ended up getting an amazing little \$20 time client call Domain Time II from www.greyscale.com. It appears to do a fantastic job in the VMs (<15ms error most of the time). Of course, it has to sample pretty frequently to get that value, but fortunately I have a physical NTP server (thanks!) on the LAN that will provide that capability. They have a server component, too.

You can pick a free evaluation of the software if you've not seen it and want to take a look. They did a great job.

Checking it's working

Check after a few minutes, to allow the program to connect to the Internet. If you are comfortable working with the command-line, obtain a command prompt (Start, All Programs, Accessories, Command Prompt; or [Windows-key-R](#), cmd), and simply enter the command:

```
ntpq -p
```

Note that the "-p" *must* be lower-case, and that there *must* be a space between the "ntpq" and the "-p". An alternative command which may work a little quicker is "ntpq -pn". You should get output similar to the display below. Alternatively, from the Start menu, select Meinberg, Network Time Protocol, Quick NTP status - which gives you a handy ntpq display which automatically refreshes every ten seconds. This screen-shot was taken *immediately* after starting, so the *poll* is 64, and the *reach* only "1".


```

C:\WINDOWS\system32\cmd.exe
Checking current status of NTP service with ntpq -p
  remote          refid          st t when poll reach  delay  offset  jitter
=====
-dns0.rmpic.co.u 193.62.22.74   2 u  14  64   1  24.796  22.103  2.235
+calx.pulsewidth 193.62.22.74   2 u  13  64   1  29.257  24.125  0.731
-etna.arboris.co 193.190.230.66 2 u  12  64   1  25.393  21.949  1.302
+oslo.illian.net 193.79.237.14  2 u  11  64   1  30.503  23.726  0.744
*ntp2.hro.nl     192.87.36.4    2 u  10  64   1  34.618  23.836  1.179
<Auto-Refresh every 10s --- CTRL+C to Cancel>

```

After some time running, the output might look more like this (synthesised) screen-shot:

```

C:\>ntpq -p
  remote          refid          st t when poll reach  delay  offset  jitter
=====
-utserv.mcc.ac.u 193.62.22.98   2 u 403 1024 377 30.433  3.186  8.331
*linnaeus.inf.ed 129.215.64.233 3 u  13 1024 377 36.255  1.329  5.719
-dnscache-london 192.36.144.22  2 u  313 1024 377 24.135  2.389  1.203
-hydrogen.dgeb.i 130.159.196.118 3 u 464 1024 377 26.581  2.736  0.896
C:\>

```

where you can see that one server (linnaeus) has an asterisk (*) as the first column, and a reach of 377. This likely indicates that all is well.

Interpreting the ntpq output:

- The display is a list of remote servers with various status reports arranged in columns.
- One **remote** server should have an asterisk (*) in the first column. This marks the server which NTP has selected as the current preferred source.
- Servers which have a plus sign (+) are good enough for NTP to sync to, others are not.
- The **reach** column should not be 0, and will expand from 1 during the normal working of NTP until it reaches 377. It is an octal display of a bit-mask showing when the server was reached. In the screenshot, NTP has been running for less than a minute, and hence only made contact once with each server. Normally you expect to see 377 in this column against each server. A column of all zeros means that NTP can't contact any servers - check your firewall settings.
- The **offset** shows how far your PC is off from a nominal UTC, and the value is in milliseconds. So the PC above is within about 1/40s of correct time!
- The **poll** value should gradually increase from 64 seconds to 1024 seconds as NTP needs to contact the server less and less frequently as the clock offset and frequency are gradually corrected. Changing the poll is automatic in NTP.
- The **delay** shows the time for a packet from your PC to reach the remote server and vice versa. Values above 150ms may indicate a satellite circuit and it's best to avoid such servers if possible. You will get best performance from servers which are close to you on the network.
- The **jitter** column shows how stable the connection between you and the remote server is.
- The **st** column shows the stratum of the server, with stratum 1 servers having a local reference such as an atomic clock or, for many servers, a radio-clock or GPS receiver reference. Most servers you will see are at stratum 2, so they are locked to a stratum-1 server. A lightly loaded stratum-2 server is probably a better reference than a heavily loaded stratum-1 server such as those with widely-publicised addresses.

Meinberg provide further information [here](#). Once basic operation is OK, if you wish to experiment I recommend you [update to a more recent version](#), as the one in the Meinberg install is the "stable" release, but the "development" versions have some performance and operational advantages.

Troubleshooting

NTP is a network application, so the basics of troubleshooting any network application apply. In today's Windows environments:

Check firewall access.

- Are the programs ntpd.exe and ntpq.exe "known" to the firewall?
- Do they have both incoming and outgoing access enabled?
- In Zone Alarm, this means "Act as a server".
- Are the access requests from the firewall software enabled - you aren't missing any?

Can you PING and TRACERT to the ntp server in question?

- Be aware that this is a poor test today, as many networks and servers disable PING access.

Can you use the "ntpq -p" command as shown above against a server your PC is trying to use?

- Again, this can be a poor test, but I found one server which responds: louie.udel.edu, however, many servers now prevent such access.

```
C:\>ntpq -p louie.udel.edu
      remote           refid          st t when poll reach  delay  offset  jitter
-----
-huey.udel.edu      128.4.1.1        2 s  518 1024 376   0.792   0.255   0.653
-dogbert.eecis.u    .GPS1.           1 s  380 1024 377   0.156  -7.678   3.710
-asok.eecis.udel    128.4.1.1        2 s  533 1024 377   1.275  -1.219   0.865
*rackety.udel.ed    .PPS.            1 u  807 1024 377   0.349  -0.509   0.359
-pogo.udel.edu      .PPS.            1 u  258 1024 377   0.469  -1.204   0.840
+ntp1.nss.udel.e    .GPS.            1 u  712 1024 377   0.405   1.242   0.407
  cdma-2.mit.edu    .INIT.           16 u   - 1024   0   0.000   0.000   0.000
-gps1.tns.its.ps    .GPS.            1 u  475 1024 377  14.683  4.850   4.670
+tick.usno.navy.    .PTP.            1 u 1045 1024 377  20.393   1.182   0.981
C:\>
```

- Do you get a similar result running "ntpq -p" on its own?

Check that you are allowed access to the server -many stratum-1 servers require that you contact them before accessing. Look for an "open access" server. There is a list of servers [here](#).

Note that NTP has a maximum clock error it will correct of 500ppm (43.2 seconds per day, or 1.8s per hour), so if you see the offset gradually increasing by more than this rate you know that your PC's clock is too inaccurate to be disciplined by NTP. But there is a work-around reported by one user - set up a scheduled task to run twice (or more often) a day to stop and restart the NTP service. Cunning! You can also [tell NTP what your clock drift rate](#) is and that may help.

For more complex issues, be aware of <http://support.ntp.org> and its "Support" web - it's a wiki, you may add or change content there.

Service will not start

This may be due to the lack of one or more DLLs

- The Microsoft Visual C++ 2008 (x86) runtime library is required. Most often this will be installed on your PC, but if not, download and install the [runtime library](#).
- If you have chosen *not* to install the SSL components above (which was not what I recommended), the service will not start as these components *are* required. Simply run the install again, selecting the "Update files" option.

I get "System error 5 - access is denied" when starting NTP

If you use the Meinberg menus to stop, start or restart NTP on Windows Vista, Windows-7 or later, you will likely get the error message above. This is because administrator privilege is needed to stop and start services. The solution is to right-click the menu item, and select "Run as administrator".

I get a message like "no associations found" from running ntpq -p

Check that your configuration file (C:\Tools\NTP\etc\ntp.conf) does contain server entries, such as in this sample simplified configuration file:

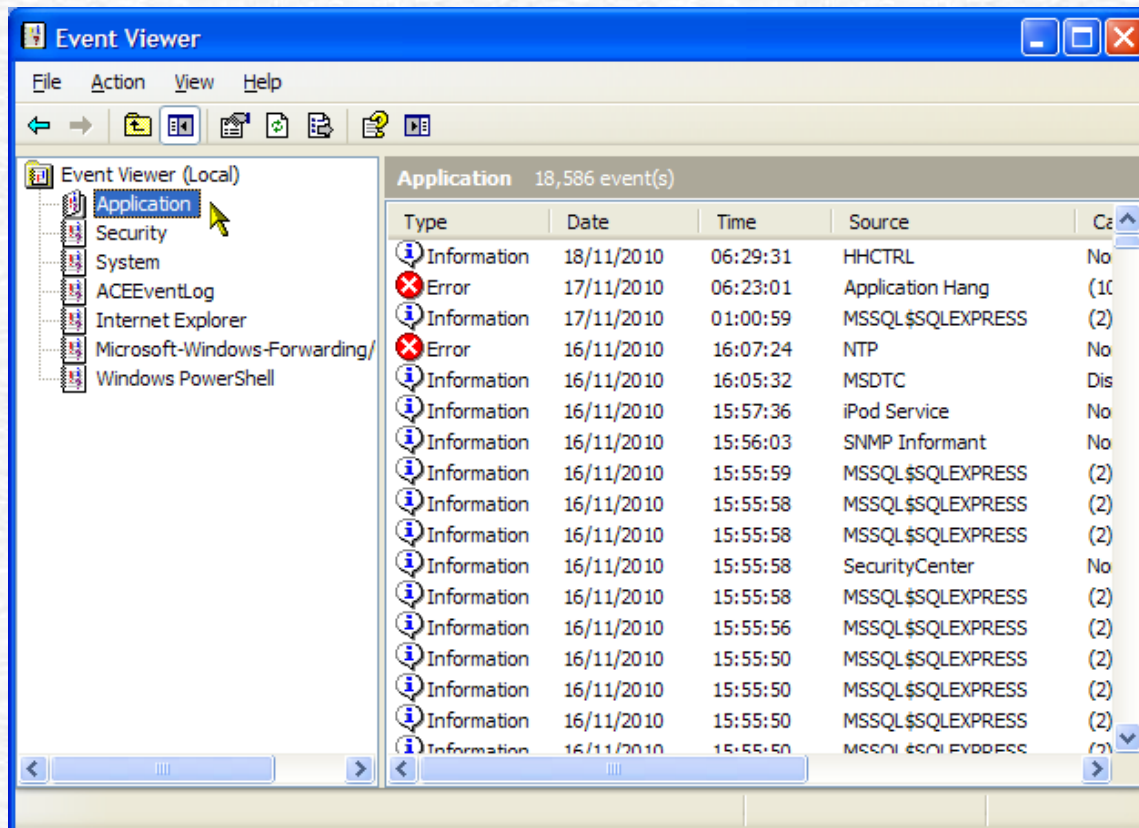
```
# Use drift file
driftfile "C:\Tools\NTP\etc\ntp.drift"
```

```
# Use NTP servers from the ntp pool project (see http://www.pool.ntp.org)
server 0.uk.pool.ntp.org iburst
server 1.uk.pool.ntp.org iburst
server 0.nl.pool.ntp.org iburst
server 1.nl.pool.ntp.org iburst
```

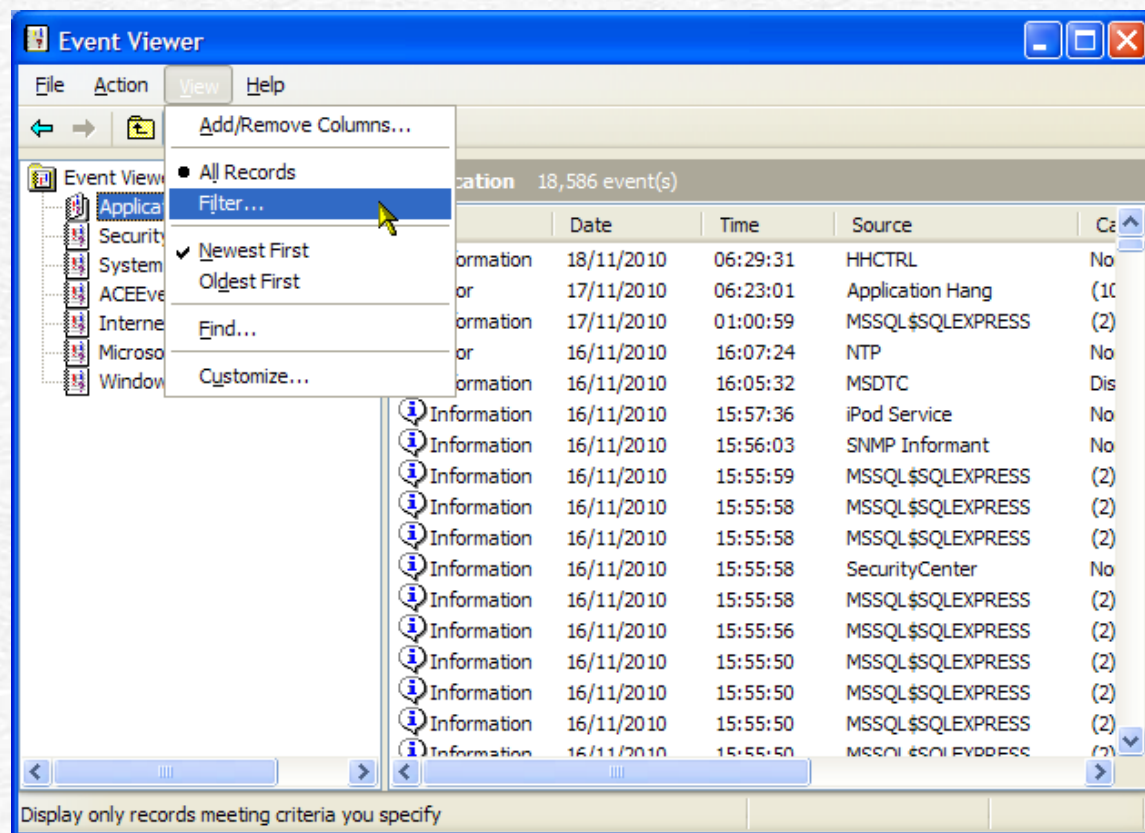
Use the Event Viewer

Windows records events from programs including NTP in the Event Log. You can use the Windows Event Viewer to look at events which NTP records as a help in diagnosing more difficult issues. There is more information about the Event Viewer for [Windows XP](#), and [Windows Vista](#) and [Windows-7](#) as the format is different between Windows versions.

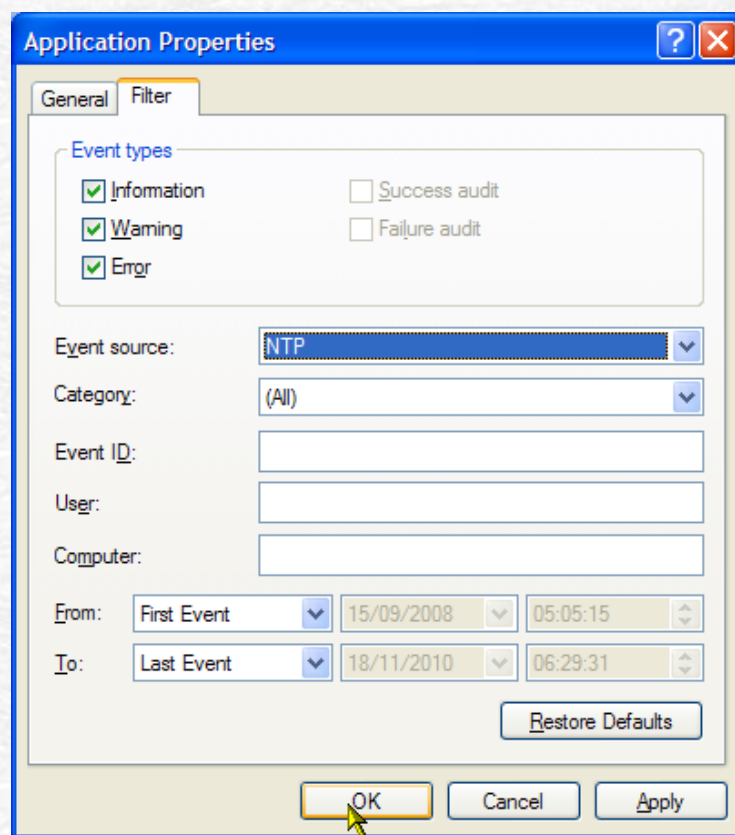
Look at the Application log by clicking on "Application":



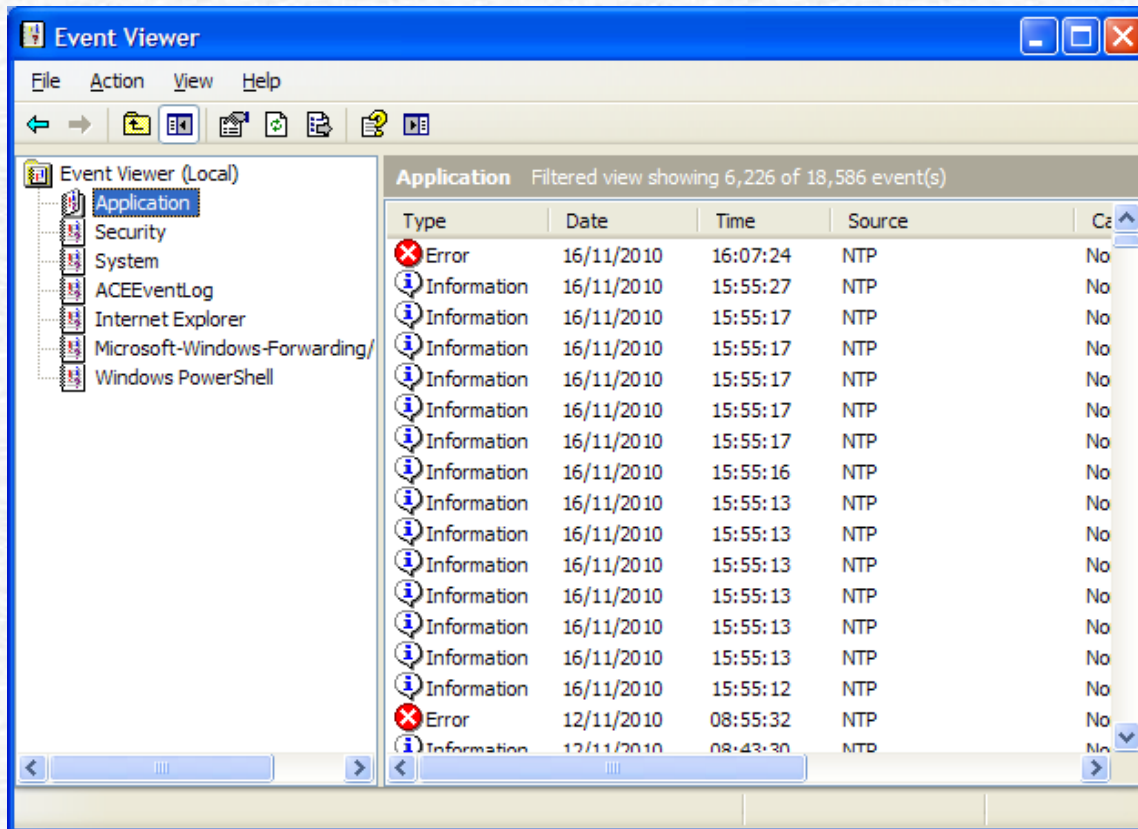
You will see a list of events from all sources, not just NTP, with the most recent events at the top. Now use the View | Filter... menu of the Event Viewer to select only those events from NTP.



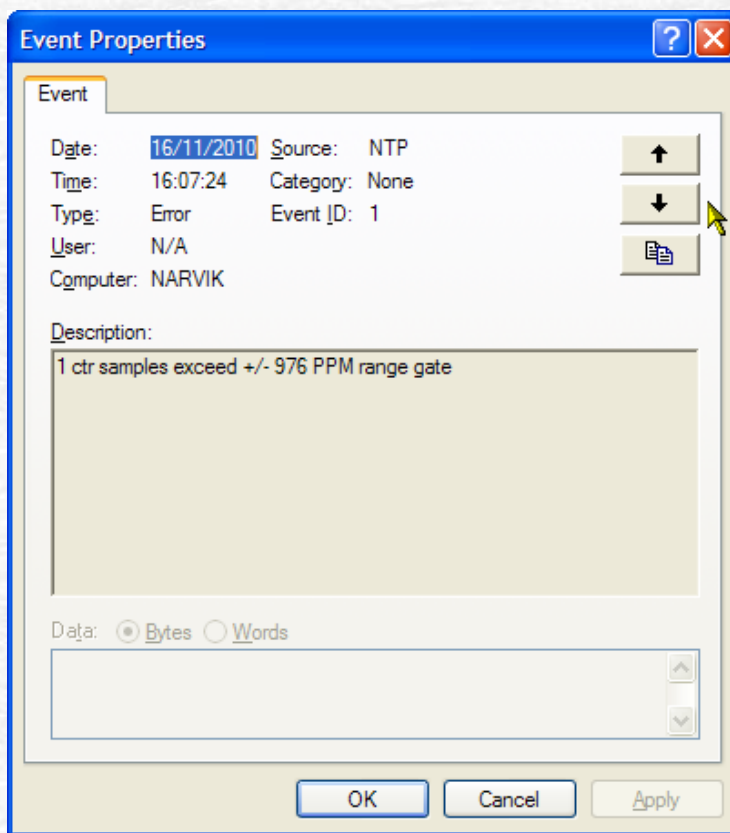
In the event source, select NTP from the drop-down list, and click the OK button.



Now double-click on one of the NTP events, in particular those with an Error status and the red cross.



and see what the actual NTP message is, for example:



You can use the up and down arrows to scroll through the NTP messages.

There is more information about using the Event Viewer on the Internet, for example, [here](#). You will get some event information messages as NTP starts and these are quite normal, and you will get occasional messages during

the normal running of NTP. Look out for those which indicate failure.

Use the Administrative Tools, Services program

You can check that the NTP service is running and, perhaps more importantly that the W32Time (the older Windows time service) is stopped using the Services applet. This may be found under the Control Panel, Administrative Tools options with Windows XP.

Check that:

- The service "Network Time Protocol" is shown as Status: Started, and Startup: Automatic.
- The service "Windows Time" is shown as Status: (blank), and Startup: Disabled

Broken PC clock

A clock near the 43 seconds/day limit

If the file ntp.drift in your `..\ntp\etc\` directory consistently shows a value near +/- 500, and NTP stops working, then it may be that your PC's clock is outside the normal limits for NTP (a drift of up to +/- 43 seconds per day is allowed). A new feature in ntp-4.2.6p4 and later allows you to provide a base offset value to NTP when it starts, so that the +/- 500 ppm (parts per million) applies to that base value, rather than being an absolute limit. So your clock must be well behaved, but need not be quite so close to correct. There are two steps to using this new functionality:

- Set the system environment variable: `NTPD_TICKADJ_PPM` to the value you need. On Windows XP, this is through the Control Panel. System..., System, Advanced..., Environment variables button, System variables. Add a New system variable, with the name `NTPD_TICKADJ_PPM` and the value 500 (or -500 if your ntp.drift was limiting at its negative extreme).
- Install ntp-4.2.6p4 or later. You can download the new version here: <http://davehart.net/ntp/win/x86/ntp-4.2.6p4-win-x86-bin.zip>
If you already have ntp-4.2.6p5 from a Sep-2012 Meinberg installation there is no need to update. Right-click, Properties, Unblock the download. Then, using the Services application, stop NTP (Network Time Protocol), save the .EXE files from your `..\ntp\bin\` directory (in case you need to revert), and copy the .EXE files from the Zip archive you downloaded. Restart NTP in Services, and check for correct operation. As the files have changed, you may need to tell your firewall to allow the new programs incoming and outgoing access.

You may need to repeat the set environment variable/restart NTP loop a number of times to get the best value for your system. Whilst I haven't tried this myself, I guess you would try and get a value in the ntp.drift file of, say, under 100, after an hour or two of operation. Have patience!

Power-saving features may cause problems

It's also possible that your PC clock appears broken to NTP, while the actual issue is some software which is tampering with the CPU speed. If you have some power-saving features enabled, you may try disabling those features and see whether NTP's performance improves. Symptoms may include the PC keeping time correctly until some CPU-intensive activity is commenced.

Other issues

- If NTP doesn't appear to start, use the Start NTP entry which Meinberg provides for you in the Start button, All Programs, Meinberg, Network Time Protocol, Service Control menu set. You may then see messages such as "Unable to log on", and then check the [event viewer](#) for more details.
- Once you have defined the password for the NTP account, the setup procedure doesn't allow you to change

it. Keep using the original password.

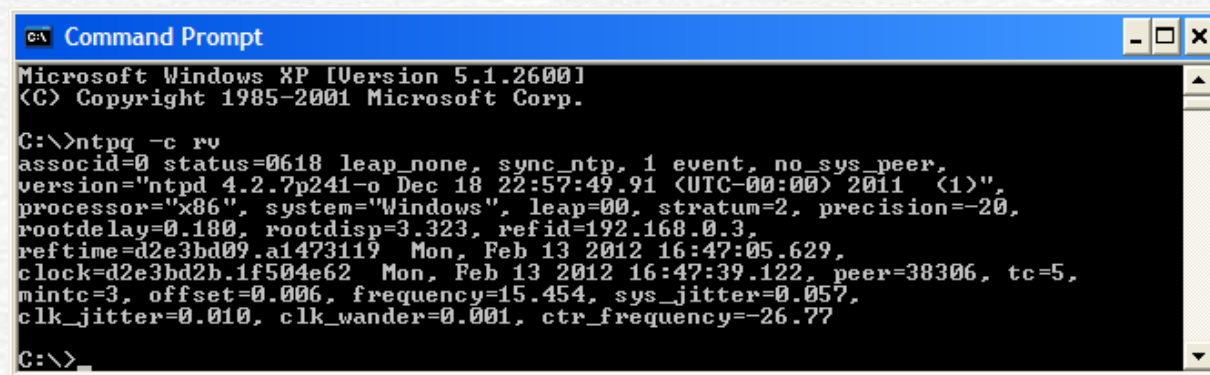
Hartmut Schulla reports: On Windows Server 2008, 64-bit, release 2, I needed to run ntpd.exe as Administrator. Right click and go to properties of ntpd.exe. Select compatibility tab and check "Run as Administrator" for all users. In the service settings select "local system account" instead of the special account created by setup. It seems for me, only "local system account" has necessary rights to access "windows time".

ntpq -p is further not working in Server2008R2 -> timeout. I don't understand why but "NTP Time Server Monitor" as replacement for ntpq from the Meinberg website works fine. Also "NTP Time Server Monitor" requires "run as administrator".

Updating NTP to a more recent version

The Meinberg site uses NTP 4.2.6p5, which is current and works well. The command:

```
ntpq -crv
```



```

C:\>ntpq -c rv
associd=0 status=0618 leap_none, sync_ntp, 1 event, no_sys_peer,
version="ntpd 4.2.7p241-o Dec 18 22:57:49.91 (UTC-00:00) 2011 (1)",
processor="x86", system="Windows", leap=00, stratum=2, precision=-20,
rootdelay=0.180, rootdisp=3.323, refid=192.168.0.3,
reftime=d2e3bd09.a1473119 Mon, Feb 13 2012 16:47:05.629,
clock=d2e3bd2b.1f504e62 Mon, Feb 13 2012 16:47:39.122, peer=38306, tc=5,
mintc=3, offset=0.006, frequency=15.454, sys_jitter=0.057,
clk_jitter=0.010, clk_wander=0.001, ctr_frequency=-26.77
C:\>

```

will include a list of parameters, including the version information on the second line.

If you do want to experiment:

1. Copy (not Move!) all the .exe files in ..\NTP\bin\ to a fresh directory for safe keeping, just in case you need to revert.
2. Download and save the new version (I suggest the latest as I will have tested that version) from [this Web page](#).
3. Right-click the Zip file, Properties, click the Unblock button if it is present.
4. Stop NTP with the Administrative Tools, Services program. It may be shown as Network Time Protocol Daemon.
5. Copy the files from the Zip download on top of those in ..\NTP\bin\
6. Restart NTP.

Any problems and you simply reverse the steps and use the version you saved in step (1). The later versions of NTP offer more options, and fix a few bugs. If you want the most recent version- [see below](#).

Using the "pool" directive

With NTP 4.2.6p5 (as installed by a recent Meinberg port), you can make a saving on the lines in the ntp.conf file, and get access to what NTP considers to be the most appropriate number of servers by replacing the lines:

```

# Use pool NTP servers
server 0.uk.pool.ntp.org iburst
server 1.uk.pool.ntp.org iburst
server 2.uk.pool.ntp.org iburst
server 0.nl.pool.ntp.org iburst

```



```
server 1.nl.pool.ntp.org iburst
```

with the single line:

```
# Use pool NTP servers
pool uk.pool.ntp.org iburst
```

NTP will determine how many servers it should connect to automatically, and it may be more than the few from a default installation. Perhaps of more significance is that NTP monitors the connectivity to each server in a "pool", and if a server goes down it will try to connect a different one, thus maintaining a good number of working servers. The pool directive introduces a new line to the ntpq -p output - don't be put off by the line showing stratum-16, this is normal, and the pool servers below that line will show normal connectivity. The top four servers are local to my network, and those below the .POOL. line are from the Internet over the WAN.

```
C:\>ntpq -p
remote          refid          st t when poll reach  delay  offset  jitter
=====
*pixie          .PPS.          1 u  15  32  377  0.128  0.243  0.010
-alta           .PPS.          1 u   9  32  377  0.171  0.281  0.072
+feenix        .GPS.          1 u   6  32  377  0.193  0.149  1.188
+stamsund      .PPS.          1 u  21  32  377  0.280  0.214  0.032
uk.pool.ntp.org .POOL.         16 p  - 1024  0  0.000  0.000  0.000
-utserv.mcc.ac.au 194.66.31.14  2 u 793 1024 377 26.383 11.823 2.144
-linnaeus.inf.ed 129.215.64.32 2 u 831 1024 377 29.435  2.716 0.485
-ns1.luns.net.uk 158.43.192.66 2 u 893 1024 377 27.221  2.439 0.633
-ntp.oceanmediag 193.190.230.66 2 u 883 1024 377 28.838  8.742 0.827
-dns1.rmpic.co.u 195.66.241.2  2 u 831 1024 377 18.708  2.299 0.744

C:\>
```

Recipe for building NTP from the source

Some notes on rebuilding NTP from the source code. Likely to be peculiar to my system and possibly incomplete. Typically, you will *not* need to recompile the source, so please treat this section as guidance and notes *only*, and *not* bullet-proof step-by-step instructions. Good luck!

Requirements:

- a compiler such as [Microsoft Visual C++ 2010 Express](#)
- the source code, from <http://www.ntp.org/downloads.html>
- [Optional] - 7-zip for Windows, from: <http://www.7-zip.org/download.html>
- a version of OpenSSL - <http://slproweb.com/products/Win32OpenSSL.html>
- ... such as OpenSSL version [1.0.0c](#) (local copy)

What do I install?

Start by downloading the Visual C++ compiler. It may be better to download the entire Visual Studio 2010 Express CD (as an ISO file) as you might want to re-install later, when Microsoft have withdrawn this version, or you might want to try some of the other components. Install the compiler to the default location, unless you are tight on space on the system drive.

Next, download and install OpenSSL. You need the full 16.9 MB installation so that you can get the source files for compiling programs to run with OpenSSL. However, you *don't* need the full OpenSSL installed as such for compiling, you only need the files for software development, so when asked whether the binaries should be copied to the Windows system directory, answer "no". After installation, look to see what environment variables have been set for OpenSSL. On a second install on a different PC, I did not see these set, and had to set them manually. You need at least two, and on my first system these were:

```
OPENSSL_INC=D:\Tools\openssl\OpenSSL.1.0.0c\inc32
OPENSSL_LIB=D:\Tools\openssl\OpenSSL.1.0.0c\out32dll
```

On the second PC, I installed to C:\Tools\OpenSSL and required the following:

```
OPENSSL_INC=C:\Tools\OpenSSL\lib
OPENSSL_LIB=C:\Tools\OpenSSL\bin
```

The compile will look for the OpenSSL files by referencing these environment variables. The %OPENSSL_INC% directory contains a sub-directory "openssl", which contains files starting "aes.h". Locate that directory and take a copy. The %OPENSSL_LIB% directory contains files with "aep.dll" as the first or second file. What I did on my first PC was to copy the files from the as-installed OpenSSL directories into the directories which I already had from a previous installation.

On the first PC, the NTP compile will also require a couple of library definition files from the OpenSSL install to be in the out32dll directory. Locate the files: libeay32.lib and ssleay32.lib in the OpenSSL install tree, and copy them to your out32dll directory. Then take a backup of that directory as well. Once you have self-contained, populated %OPENSSL_INC% and %OPENSSL_LIB% directories, you can uninstall OpenSSL. I suspect that my lack of C/C++ experience may have led me to make a poor choice for either the install directory or the %OPENSSL_LIB% environment variable.

Finally you can download the source code. It will be in the form of a 6 MB UNIX zipped archive, such as: ntp-dev-4.2.7p326.tar.gz. I find that the easiest way to unzip this is to send it to the 7-zip File Manager, where it appears as a file of unknown properties, double-click on that file and it turns into a directory within the 7-Zip File Manager, and then just drag that directory on to your desired location in an open Windows Explorer window. In my case, I wanted to put the source into D:\Tools\, so after the drag-and-drop I ended up with a directory:

```
D:\Tools\ntp-dev\ntp-dev-4.2.7p326\
```

Compiling

The NTP source comes ready to compile in Visual Studio 2005, 2008 and 2013, but not 2010. However, this is no problem, as VS 2010 will convert the .SLN (solution) file into the new format. Locate the file:

```
D:\Tools\ntp-dev\ntp-dev-4.2.7p326\ports\winnt\vs2008\ntp.sln
```

and double-click on it. If you have more than one VS2010 product installed, you may need to right-click on the ntp.sln file and select: Open with, Visual C++. At this point, Visual Studio will invite you to convert the project, so allow it to do so. There should be no errors. On the top line of Visual Studio there are three drop-down lists. On the first drop-down, select "Release" (chooses between making a Release or a Debug version). The second should be "Win32", and the third will depend on your last search. Then, from the Debug menu, select Build Solution (or press F7 - being a Delphi man I am more used to pressing F9!). After a few minutes (it would be a few seconds in Delphi), you should get a message "==" Build: 9 succeeded, 0 failed, 0 up-to-date, 0 skipped == or similar. If you get errors, I am the last person to ask! Sorry!

Installing the newly compiled version

The new files will have been created in:

```
D:\Tools\ntp-dev\ntp-dev-4.2.7p326\ports\winnt\vs2008\Win32-bin\Release\
```

If you can't compile NTP yourself, there are downloadable sets of recent, ready-made .exe files [here](#). Be sure to right-click, Properties, Unblock the .zip file after downloading, and then copy the files into a temporary directory.

It may be most convenient to sort that directory by Type (click on the Type column header), and you will see 6 .EXE files and one .DLL. There may be more or fewer .EXE files. My own practice is to:

- Use the Services manager to stop NTP
- Copy the new ntp*.exe files from the Release to my ..\ntp\bin\ directory
- Use the Services manager to start NTP

In fact, I do this so much that I made a batch file for the task, which needs to be run as administrator in Windows Vista and later. The core of the file is:


```
set DEST="C:\Tools\NTP\bin\  
net stop ntp  
copy ntpd.exe %DEST%  
copy ntpdate.exe %DEST%  
copy ntpdc.exe %DEST%  
copy ntp-keygen.exe %DEST%  
copy ntpq.exe %DEST%  
net start ntp  
timeout 20  
ntpq -crv -pn
```

I've coded it this way - with the %DEST% - because I have a similar file on each of my systems which run NTP. I can set DEST according to the needs of the particular PC, leaving the bulk of the file unchanged. In practice, providing you are staying within the same major version (i.e. 4.2.7), copying the ntpd.exe alone is likely to be sufficient. My actual file has some logging as well. The ntpq call at the end of the file provides a check that the update has produced a working ntp, after the 20 second delay produced by the *timeout* command.

If working on Windows XP, the *timeout* command is not available, and can be replaced by:

```
ping -n 20 127.0.0.1
```

OpenSSL version issue

Earlier versions of the Meinberg NTP were distributed with OpenSSL 0.9.8, and later versions with 1.1.0c. However, we have found that if you compile with 0.9.8k the resulting NTP executable files will not run on a system with the 1.1.0c DLLs, and vice-versa. The working compromise appears to be to get version 1.0.0c OpenSSL, as the resulting .EXE files will run on systems with either OpenSSL DLLs. Thanks to Dave Hart, I managed to locate a copy of this version, which I have placed [here](#).

A catch for Windows 2000

Programs compiled with VS 2010 don't run on Windows 2000.

Fix needed for NTP 4.2.7p349-356

Between NTP 4.2.7p349 to 4.2.7p356, a small edit was required to two of the files which were incorrectly automatically generated. The fixes are described [here](#).

Why NTP?

People sometimes ask why they should use this NTP software, rather than the W32Time service built into Windows, or other so-called "atomic time" programs.

- NTP uses multiple servers, and automatically chooses the best one based on continuous monitoring. If a server goes down, another one is selected without user intervention. By comparison W32Time, like many clock programs, only uses a single server.
- NTP contacts the servers only as frequently as needed, between once a minute and once every 17 minutes. W32Time, by default, only contacts servers once a week, so your PC could be minutes out between corrections!
- NTP from the Meinberg installation will try and use [pool servers](#) located nearer to you for better performance, and not servers located a continent away.
- NTP runs as a service once setup - it will even keep time while you are logged off your PC.
- NTP is much more accurate:
 - NTP alters the clock rate, as well as initially setting the clock time, so that the clock will stay as close as possible to UTC between corrections.
 - Once NTP is running, the time is not routinely stepped, but is adjusted smoothly.

- Time will normally not step backwards.
- Leap seconds are handled without user intervention, and without stepping the PC's clock.
- Even on Windows, accuracies down to the millisecond level can be achieved with suitable configurations - required for certain software.

- You can use your own time sources with NTP, such as a radio-clock or [GPS receiver](#), and it will then work independently of an internet connection, such as when you are out in the field.
- You can run NTP on many operating systems including Windows, Linux, FreeBSD and the Mac OS, so managing and monitoring NTP expertise doesn't need to be re-learned.
- NTP provides comprehensive performance checks, other programs typically do not. Monitoring can be across the network from a central point, if needed. Here is an example of [automated NTP monitoring](#).
- NTP is open-source software, 100% compliant with the Internet protocol for network time.

The Next Step forward - your own precision time server

Having appreciated the benefits of installing NTP, you may wonder about keeping precise time when the Internet goes down, or on a PC without an Internet connection, or perhaps you want your PC(s) to keep even closer time than NTP provides as standard? The answer is to setup your own Stratum-1 reference server, and you can do this either using Windows, Linux or [FreeBSD](#). No extra software is required as the NTP install can act as both a client to a remote server, or as a server itself to local PC clients. NTP is a very small load, so you can run it as an extra on an existing PC without impacting the work which may be running there. All you need is a GPS device which outputs a PPS (pulse per second) signal, and these are usually accurate to within a microsecond even on the lowest cost of today's GPS timing devices. For a low-cost device, I recommend the [Sure Electronics GPS evaluation board](#), and I have written up a page on [how to use the Sure board](#). This costs about £25 (US \$35) including the magnetic puck antenna and USB lead, so it is unlikely to break the bank! There are alternatives listed on my [Raspberry Pi NTP server](#) page, and you will have fun making a Raspberry Pi NTP server!

Once you have one accurate PC within your network, you can sync the other PCs to that either over the LAN or with a wireless connection, and all PCs will have improved accuracy compared to an Internet-only synced configuration. Note that you may need to set your firewall software so that PCs other than your main server can access NTP - set NTP to allow inbound access as described [here](#). To test, be sure that you can run: `ntpq -pn <my-time-server>` from a client PC. If you get a time-out response, likely it's a firewall issue. When the PCs are only talking to LAN servers, you can make them even more precise by polling your local reference clock more frequently, and lessen your load on the Internet servers by polling them less frequently, so you might use configuration file such as:

```
# Use local stratum 1 server
server my-time-server iburst maxpoll 5 prefer

# Use pool NTP servers
pool uk.pool.ntp.org minpoll 10 iburst
```

Replace `my-time-server` by the name or IP address of your local accurate PC. A configuration such as this makes the PC poll the local stratum-1 server every 32 seconds ($2^{*}5$), but only poll the Internet servers once every 1024 seconds ($2^{*}10$). You should keep the Internet servers as a backup, though, in case your local NTP server fails or loses GPS lock or whatever. Why not make this your next project?

Running a publicly accessible NTP server

If you are running a server which is accessible from the public Internet - perhaps you are contributing to the [NTP Pool](#) project - there are some simple precautions you should take to ensure that your server is not used as the source of an attack on other PCs. Note that this doesn't apply to most end-user clients sitting on your local PC, you would need to have specially opened a port in your firewall or router to allow public incoming unsolicited UDP port 123 packets into your local network. If you are using a development version (4.2.7p26 or later) you are already protected. The following notice explains more:

NTP users are strongly urged to take immediate action to ensure that their NTP daemon is not susceptible to use in a reflected denial-of-service (DRDoS) attack. Please see the [NTP Security Notice](#) for vulnerability and mitigation details, and the [Network Time Foundation Blog](#) for more information. (January 2014)

Copyright © [David Taylor](#), Edinburgh



Last modified: 2014 Jul 13 at 11:01