

**Tardis 2000 v1.6**

# 1 Contents

<b>1 CONTENTS</b>	<b>2</b>
<b>2 SHAREWARE</b>	<b>4</b>
<b>3 WELCOME TO TARDIS 2000</b>	<b>5</b>
<b>4 INSTALLATION</b>	<b>6</b>
4.1 TARDIS 2000	6
4.2 APPLICATION OR SERVICE?	6
4.3 INSTALL COMMAND LINE OPTIONS	6
4.4 MANUAL INSTALLATION	7
4.5 RUNNING TARDIS 2000 SERVICE IN DEBUG MODE	8
4.6 COEXISTENCE OF TARDIS 2000 SERVICE WITH WINDOWS TIME SERVICE	8
4.7 LOGGING	9
4.8 TARDIS 2000 AND WINDOWS FIREWALL	9
<b>5 EXAMPLES OF USING TARDIS 2000</b>	<b>12</b>
5.1 A STAND-ALONE PC USING A DIAL-UP CONNECTION TO THE INTERNET	12
5.2 A PC ON A LAN ACCESSING INTERNET SERVICES	12
5.3 A PC ON A LAN USING LOCAL TIME SERVERS	12
5.4 A PC WITH A GPS OR RADIO DEVICE	12
5.5 TARDIS 2000 ACTING AS A TIME SERVER	12
<b>6 CONFIGURING TARDIS</b>	<b>13</b>
6.1 MAIN SCREEN	13
<b>7 OPTIONS PAGES</b>	<b>16</b>
7.1 SETTING THE TIME	16
7.2 DIALUP	18
7.3 GENERAL OPTIONS	19
7.4 INFORMATION PAGE	20
7.5 GPS	22
7.6 BROADCAST NTP/NTP	23
7.7 HTTP PROXY SETTINGS	25
7.8 ALERTS	25
<b>8 TIME SERVERS</b>	<b>27</b>
<b>9 TECHNICAL SUPPORT FOR TARDIS 2000</b>	<b>28</b>
<b>10 REGISTERING AND PAYING FOR TARDIS 2000</b>	<b>29</b>
10.1 PAYMENTS	29
10.2 UPGRADE POLICY	30
<b>11 FREQUENTLY ASKED QUESTIONS</b>	<b>34</b>
<b>12 TIME PROTOCOLS</b>	<b>40</b>

<b>12.1 NTP BROADCAST PROTOCOL</b>	<b>40</b>
<b>12.2 NMEA</b>	<b>40</b>
<b>12.3 HTTP PROTOCOL</b>	<b>40</b>
<b>12.4 GALLEON RADIO CLOCK</b>	<b>41</b>
<b>12.5 KALLISTO GPS CARD</b>	<b>42</b>
<b>12.6 RFC2030 (SNTP)</b>	<b>44</b>
<b>12.7 RFC867 (DAYTIME)</b>	<b>56</b>
<b>12.8 RFC868 (TIME)</b>	<b>57</b>
<b>12.9 RFC792 (ICMP TIMESTAMP SECTION)</b>	<b>58</b>
<b>12.10 OTHER RADIO AND GPS CLOCKS SUPPORTED.</b>	<b>60</b>

## **2 Shareware**

### Copyright Notice

1994-2015 H.C. Mingham-Smith Ltd. ("The author")

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Tardis 2000 is Shareware. This is a complete working version. There are no annoying reminder screens about what it costs, and there are no disabled features. If you continue to use it after evaluating it please send payment as detailed on page 29.

### 3 Welcome to Tardis 2000

Tardis 2000 is a utility for Windows that makes sure your PC's clock tells the right time. It can find out what the right time is in various ways including using networked timeservers, GPS (The Global Positioning System), Radio clocks, and by listening for time broadcasts over a LAN. Tardis 2000 requires the TCP/IP networking protocol to be installed.

The service version of Tardis 2000 runs as a Windows "service", just like other services that come as standard with Windows.

It acts as both a server and client for the supported protocols. I.e. it can get the time from a timeserver with the 'correct' time and then make the 'correct' time available to local clients.

Tardis 2000 running on a central server can be used as a master time source for the domain by running Tardis or K9 on the other workstations of the domain using the server machine as the timeserver. See <http://www.mingham-smith.com/> for details of K9.

Tardis 2000 can use the following time protocols

- RFC868 (Time)
- RFC867 (Daytime)
- RFC2030 (SNTP)
- Broadcast NTP
- HTTP protocol.
- Kallisto GPS.
- NMEA compliant GPS cards.
- Galleon Radio Clock
- ICMP timestamp
- Trimble Palisade
- EndRun Technologies Præcis Ct
- Kinometrics Truetime
- Spectracom WWVB
- NeoClock
- Expert mouseCLOCK
- Serial Radio Clock connected to DCD pin
- EMC Professional in XNTP mode
- IRIG-B

#### Thanks

We would like to thank Mark Symons who helped enormously in the development of Tardis 2000.

## 4 Installation

### 4.1 Tardis 2000

Tardis 2000 is distributed as a setup file called tardis2000.exe. This manual and the installable Tardis 2000 files are contained within it. Run the *tardis2000.exe* program and Tardis 2000 will automatically be installed.

### 4.2 Application or Service?

If your PC runs Windows 95/98/ME you will only be given the option to install the Application version of Tardis. If you use more recent versions of Windows you have the option of using either the application version or the service version.

We recommend that you use the service version.

Both versions have essentially the same functionality. The difference is that the application is a stand-alone utility; the service version is a control panel and a background service that runs all the time whether anyone is logged in or not.

You should log into your Windows NT/2000/XP/2003 system as a user with administrative privileges. Tardis 2000 must be installed and configured by someone with administrative privileges.

### 4.3 Install command line options

Tardis 2000 now uses the wonderful Inno Setup (<http://www.jrsoftware.org/isinfo.php>). As a result the tardis2000.exe setup program has the following install options provided by Inno setup.

The tardis2000.exe Setup program accepts optional command line parameters. These can be useful to system administrators, and to other programs calling the Setup program.

<b>/SP-</b>	Disables the <i>This will install... Do you wish to continue?</i> prompt at the beginning of Setup.
<b>/SILENT, /VERYSILENT</b>	<p>Instructs Setup to be silent or very silent. When Setup is silent the wizard and the background window are not displayed but the installation progress window is. When a setup is very silent this installation progress window is not displayed. Everything else is normal so for example error messages during installation are displayed and the startup prompt is (if you haven't disabled it the '/SP-' command line option explained above)</p> <p>If a restart is necessary and the '/NORESTART' command isn't used (see below) and Setup is silent, it will display a <i>Reboot now?</i> message box. If it's very silent it will reboot without asking.</p>
<b>/LOG</b>	<p>Causes Setup to create a log file in the user's TEMP directory detailing file installation actions taken during the installation process. This can be a helpful debugging aid. For example, if you suspect a file isn't being replaced when you believe it should be (or vice versa), the log file will tell you if the file was really skipped, and why.</p> <p>The log file is created with a unique name based on the current date. (It will not overwrite or append to existing files.) Currently, it is not possible to customize the filename.</p> <p>The information contained in the log file is technical in nature and</p>

	therefore not intended to be understandable by end users. Nor is it designed to be machine-parseable; the format of the file is subject to change without notice.
<b>/NOCANCEL</b>	Prevents the user from cancelling during the installation process, by disabling the Cancel button and ignoring clicks on the close button. Useful along with '/SILENT' or '/VERYSILENT'.
<b>/NORESTART</b>	Instructs Setup not to reboot even if it's necessary.
<b>/RESTARTEXITCODE=exit code</b>	Specifies the custom exit code that Setup is to return when a restart is needed. Useful along with '/NORESTART'. Also see Setup Exit Codes.
<b>/LOADINF="filename"</b>	Instructs Setup to load the settings from the specified file after having checked the command line. This file can be prepared using the '/SAVEINF=' command as explained below. Don't forget to use quotes if the filename contains spaces.
<b>/SAVEINF="filename"</b>	Instructs Setup to save installation settings to the specified file. Don't forget to use quotes if the filename contains spaces.
<b>/DIR="x:\dirname"</b>	Overrides the default directory name displayed on the Select Destination Location wizard page. A fully qualified pathname must be specified.
<b>/GROUP="folder name"</b>	Overrides the default folder name displayed on the Select Start Menu Folder wizard page.
<b>/NOICONS</b>	Instructs Setup to initially check the Don't create any icons check box on the Select Start Menu Folder wizard page.
<b>/COMPONENTS="comma separated list of component names"</b>	Overrides the default components settings. Using this command line parameter causes Setup to automatically select a custom type.

#### 4.4 Manual Installation

Occasionally the installation of the Tardis 2000 Service must be done manually, usually where other software interferes with the automatic installation process. The Tardis 2000 Service files can be requested by E-mail and installed manually as follows.

Log into your Windows NT/2000/XP/2003 system as a user with administrative privileges.

Decide which directory you are going to put the tardisNT.exe in, and move it there. A good choice is the \WINNT\SYSTEM32 directory, which is where many other services live. Using the Security/Permissions menu option in the File Manager, ensure that the SYSTEM user has read permission for the file.

Install tardisNT.exe as a service by running the program from the Windows NT/2000/XP/2003 command line, specifying the *add* flag. (NOTE - it is vital that you execute this command specifying the copy of tardisNT.EXE which you placed in the \WINNT\SYSTEM32 directory, and not using some other copy which you plan subsequently to delete.) For instance:

```
tardisNT add
```

The program will register itself and its location with the Service Manager, and will report success or failure. In the case of failure, see the section on Installation Problems below.

To verify that the installation has succeeded, start the Windows NT/2000/XP/2003 Control Panel and double-click on the Services icon. The resulting dialog should list Tardis as one of the installed services.

##### 4.4.1 Installation Problems

Message	Meaning
The system says that tardisNT.EXE is not a Windows NT/2000/XP/2003 program	This is probably because you are trying to run an executable for the wrong sort of processor or the file

This function is only valid in Win32 mode. (0x78)	is damaged.
“Failed to create service.”	You aren’t running Windows NT/2000/XP/2003!
Could not start tardisNT time synchronization service on \\yourmachine.Error 0002: The system cannot find the file specified.	This error message occurs if you try to install tardisNT when it is already installed.
Could not start tardisNT time synchronization service on \\yourmachine.Error 0005: Access is denied.	The Service Manager could not locate tardisNT.EXE. This probably means it has been moved, or has not been installed correctly. Remove and reinstall tardisNT - see section above for details.
	tardisNT.EXE is inaccessible to the SYSTEM user. By default, the Service Manager starts the tardisNT process running under a user ID of SYSTEM. The executable file for the service must be readable by this user.

---

#### 4.4.2 Manual Deinstallation

This section describes what to do if you want to remove the Tardis 2000 Service from your computer.

At the Windows NT/2000/XP/2003 command line, run tardisNT with the cleanup option:

```
tardisNT cleanup
```

This will remove the service and the control panel after a reboot.

#### 4.5 Running Tardis 2000 Service in debug mode

To make debugging easier you can run Tardis in debug mode like so.

```
TARDISNT debug
```

This should run Tardis as a console application. Log information and debugging are shown on the screen.

Make sure that the service isn’t running when you run tardisnt in debug mode otherwise you will be running it twice.

#### 4.6 Coexistence of Tardis 2000 service with Windows Time Service

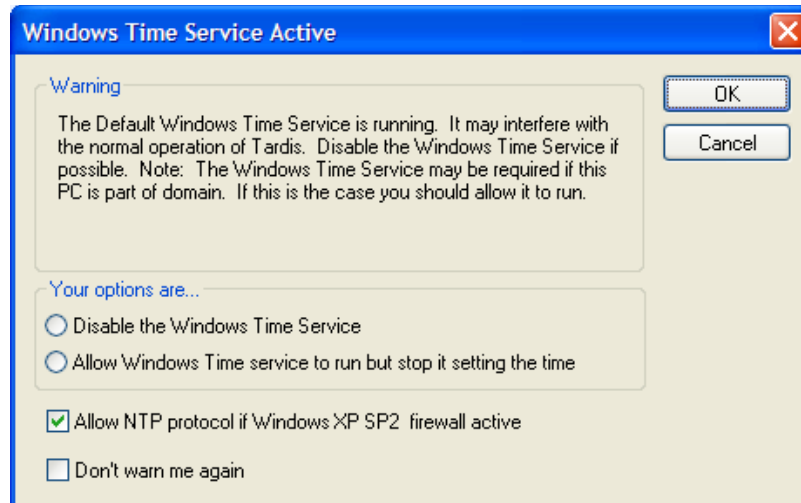
Windows 2000, XP and more recent versions have a primitive Time Service installed by default.

The Tardis 2000 Service and the Windows Time Service require some help to coexist on the same PC because they both try to use the same NTP network port.

Tardis 2000 allocates the port and the Windows Time Service complains and is generally unhappy.

When the control panel is started Tardis 2000 will warn you when the Windows Time Service is active.





You have the option to stop the Windows Time Service removing the problem. If this option is selected the Windows Time Service will be stopped and disabled. Note: This may not be possible if other services depend on the Windows Time Service, as may be the case on primary domain controllers.

The best workaround is to make the Tardis 2000 service depend on the Windows Time Service so it starts second. Tardis 2000 will work around port 123 being unavailable and everything should be ok. Tardis 2000 may not be able to act as an NTP server if this option is selected.

On Windows firewall blocks NTP requests from reaching Tardis 2000. Select the option to allow NTP or open the port manually as shown here.

#### 4.7 Logging

If an error in the operation of the service occurs, the error will be logged in the *Application* Event Log. This log may be viewed with the Event Viewer, which you will find in the Administrative Tools program group. See your Windows NT/2000/XP/2003 documentation for details of how to use the Event Viewer.

The events logged in the *Application* Event Log can be associated with a Tardis problem (e.g. a file I/O error, or a system call failure caused by lack of resources, or a problem with the configuration information).

Problems associated with the network are recorded in the *Application* Event Log as Warning events.

Tardis also logs information such as Time updates and service starts and stops.

Further information on events that may be logged by Tardis is given in the chapter on troubleshooting on page 20 of this manual.

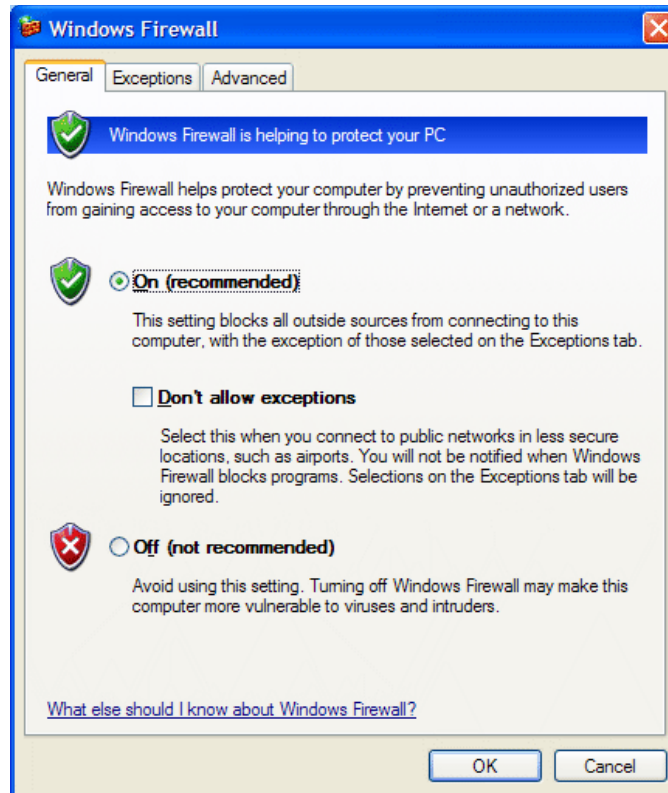
Note that the Event Viewer uses the TARDIS.CPL file to interpret messages associated with events. Therefore, if you delete the TARDIS.CPL file, Tardis events in your Application Event Log will be unintelligible.

#### 4.8 Tardis 2000 and Windows firewall

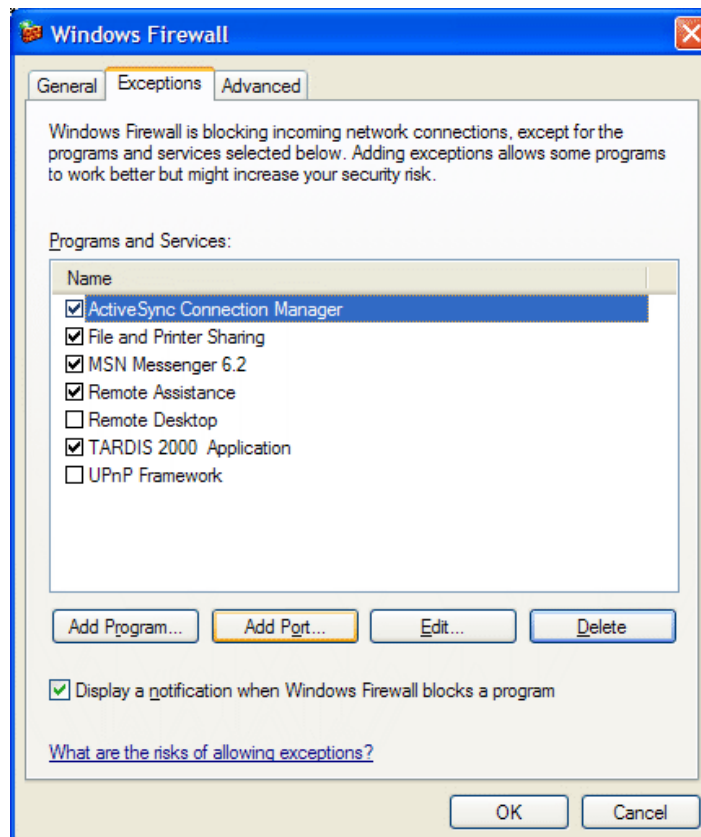
Windows a firewall feature that may interfere with the normal operation of Time synchronization products.

This describes how to configure the firewall to allow unsolicited NTP messages through to K9 and Tardis.

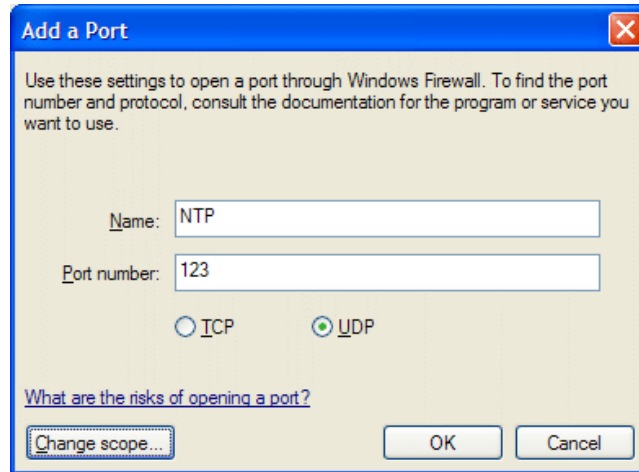
Open the firewall control panel. Make sure the settings are set like this



On the Exception tab click on 'Add Port'



Then add a setting to allow NTP.



Press OK and everything should work.

## **5 Examples of using Tardis 2000**

### **5.1 A Stand-alone PC Using a Dial-Up Connection to the Internet**

Many users configure Tardis to reset their PC's clock each time they log on to the Internet using a dial-up connection. Tardis should be running before the dial-up connection is established and configured to access a time server once you are logged on to the Internet. Simply select "Options", click on the dial-up tab, then click on "Watch for dial-up". When Tardis 2000 detects an active RAS connection it uses it to access a time server and sets the time, otherwise it will sit there waiting. If the option is not set Tardis may cause a dialup connection to be started depending on your dialup networking settings. Tardis knows nothing about this and will assume that you are on a LAN, Tardis will *not* shutdown any connection automatically started in this way.

Tardis 2000 incorporates a default set of Time Servers. It will always access the one at the top of the list, so if you wish to obtain the time from a particular server move it to the top of the list (by right clicking on it and selecting "move to top").

### **5.2 A PC on a LAN Accessing Internet Services**

For LANs connected directly to the Internet, any PC on the LAN can run Tardis 2000 as outlined above. Alternatively a single PC can run Tardis 2000 to obtain the time and then broadcast it to the other PCs on the LAN with the clients running Tardis 2000's companion program K9.

However, many LANs are connected to the Internet via firewalls which block the protocols used by Tardis 2000 and other time synchronisation programs. Firewalls are usually open to some protocols, typically the HTTP protocol, and Tardis 2000 is unique in its ability to use the HTTP protocol to obtain the time.

To use the HTTP protocol, select HTTP in the Server details dialog.

(It should be noted that other protocols might work if the firewall allows access to the relevant ports)

The situation for a proxy server is similar to a firewall. Proxy servers block all protocols except HTTP and are typically used on company Intranets.

To use a proxy server the settings must be set up in the proxy server options tab.

### **5.3 A PC on a LAN Using Local Time Servers**

Tardis 2000 can use existing time servers on a company LAN to ensure that PC clocks are synchronised locally. Typical time servers are UNIX machines, routers, dedicated time server devices, and other copies of Tardis.

### **5.4 A PC with a GPS or Radio Device**

Tardis 2000 can obtain the time from GPS and Radio devices so no Internet connection or other time server is required. If one PC on a LAN is equipped with a GPS or Radio device, Tardis 2000 can, of course, broadcast the time obtained from this to clients on the LAN (the clients running K9 as outlined above).

### **5.5 Tardis 2000 Acting as a Time Server**

By default, Tardis 2000 is always a server as well as a client so other PCs on a LAN can use it as a time source. This would ensure that the clocks of all PCs on a LAN were synchronised to the same time.

## 6 Configuring Tardis

Tardis and the NT/2000/XP/2003 service version of Tardis are slightly different in the way they are configured. The NT/2000/XP/2003 version has a control panel (Start->Settings->Control panel in NT/2000, control panel->date time in XP), the standalone version is an application in its own right. Although they are started differently the user interfaces are basically the same.



### 6.1 Main Screen

Contents of the main Tardis screen are described below.

<b>Status</b>	The status messages show what Tardis is up to. The text will change to provide useful progress and error messages. These messages can be recorded. See the section on Information.
(Not present in service version).	
<b>Add</b>	Add allows new time servers to be configured. The Server details dialog allows the details of the server to be entered. Double clicking a blank space in the server list will also add a server.
<b>Clear all</b>	is a quick way to remove all the servers. You will be asked to confirm that you want them all deleted.
<b>Import and Export</b>	allow you to load and save lists of servers. This is useful for quickly configuring Tardis with pre-set lists of servers.
<b>Default</b>	Restore the default list of timeservers

Clicking a server with the right hand mouse button gives the following options.

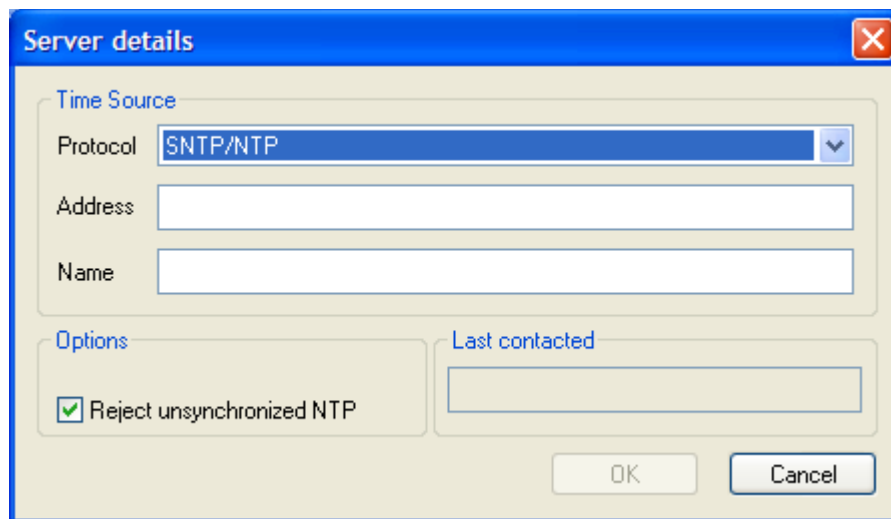
<b>Open</b>	has the same effect as double clicking the server. An attempt will be made to connect to it.
<b>Move to Top</b>	moves this server to the top of the list.
<b>Move up</b>	moves this server up one.
<b>Move Down</b>	moves this server down one.
<b>Move to Bottom</b>	moves this server to bottom of list.
<b>Delete</b>	Deletes this server. This can also be done by selecting a server and pressing the delete key.
<b>Properties</b>	will start Server details dialog to allow the server's details to be modified.



- Time Services** When the a time server is used the relevant indicator will flash. This is not present on the service version.
- Start/Stop Service** This is only present on the service version. It allows the service to be easily stopped and started.
- Options** This button allows you to configure Tardis's many features. (Not present on service version)

### 6.1.1 Server details

This dialog is used to add new servers and to edit existing time servers.



- Protocol** Select the protocol used by your timeserver here.
- SNTP is best if you have access to servers that support it. It is the standard way to synchronize computer clocks.
- RFC 868 Time protocols. These allow corrections to the nearest second. Time/udp uses very little network bandwidth.
- HTTP protocol may be required if you are using a firewall/proxy and have no timeservers on your LAN.

NTP broadcast protocol is a good choice if you have an NTP server on your LAN. It can be configured to broadcast time information. Tardis will listen for these broadcasts if you use this protocol.

Kallisto GPS option can be used to get the time from a Kallisto GPS card.

NMEA GPS supports NMEA compliant GPS devices.

The Galleon Radio Clock option can be used to get the time from a Galleon Radio Clock.

RFC867 protocols are a primitive way to get time information. This should only be used when there is no other alternative. Since the daytime protocol does not specify the format of the data returned Tardis must make some compromises. This protocol will only correct the time by up to  $\pm 30$  minutes.

ICMP Timestamp can correct the time to  $< 1$  second and has the advantage that it is 'serverless'. This protocol is so low level that it asks the server machine's TCP/IP driver software for the time rather than requiring a time server program to be running. Not all IP devices support this feature and those that do don't always do a good job of it. Linux does it properly. Later versions of windows are ok.

- Trimble Palisade
- EndRun Technologies Præcis Ct
- Kinometrics Truetime
- Spectracom WWVB
- Neol NeoClock
- Expert mouseCLOCK
- Serial Radio Clock connected to DCD pin
- EMC Professional in XNTP mode
- IRIG-B

These are all serial port connected radio or GPS clocks.

<b>Address</b>	<p>This setting is the address of the machine that knows the correct time. It may be entered as a name, e.g. tycho.usno.navy.mil, or as an Internet address e.g. 123.123.123.123.</p> <p>If you are using a clock connected to a serial port set the address to the COM port the clock is connected to e.g. COM2</p> <p>When using the NTP broadcast protocol the address may be left blank in which case Tardis will listen to any broadcasts. If an address is entered then Tardis will only listen to broadcasts from that machine.</p>
<b>Name</b>	<p>You may enter a descriptive name for the server here. If none is entered the address of the server is used instead.</p>
<b>Reject unsynchronized NTP</b>	<p>If this is set Tardis will reject time information from NTP servers that claim to be unsynchronised. This can happen if the server has lost touch with its time source.</p>
<b>Use proxy server</b>	<p>When you use the HTTP protocol you can select this to use a proxy server. This is often required where Tardis is being used within a company Intranet.</p>

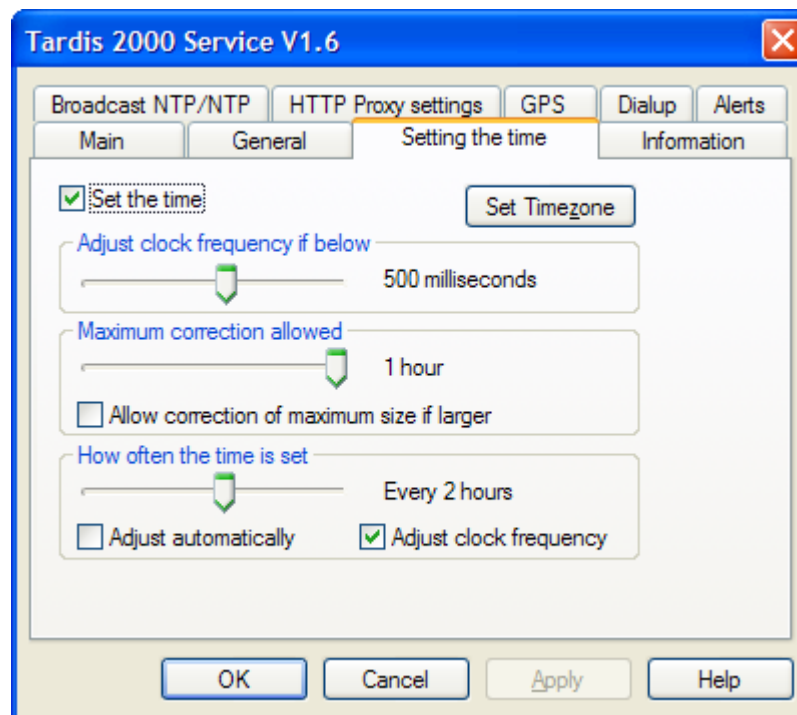
## 7 Options Pages

The options pages allow you to configure Tardis.

<b>Setting the time</b>	These options control when and if the time is set.
<b>Dialup</b>	This controls how dialup connections are handled. If you don't have RAS installed this page will be absent.
<b>General</b>	This page contains miscellaneous options.
<b>Information</b>	Statistics and logging options are set using this page.
<b>GPS</b>	If you have access to a GPS device it can be configured on this page.
<b>Broadcast NTP/NTP</b>	Configure NTP and NTP broadcasts for use with Tardis or K9.
<b>HTTP Proxy settings</b>	Set the HTTP proxy settings for use with HTTP connections.
<b>Alerts</b>	Configure Alerts to inform administrators of loss of synchronization.

### 7.1 Setting the time

These options control when and if the time is set.



<b>Set the time</b>	If this is set Tardis will set the system time, Switch it off if you don't initially trust the server you are connecting to. It gives you a chance to see what kind of time it is going to give you first without setting your PC's time to 10:61 77 Jan. 1914 accidentally.
<b>Set Timezone</b>	This button takes you to the Control Panel for Date and Time so you can set up your timezone and whether you use daylight saving time or not.
<b>Minimum</b>	Tardis checks that time correction is worth bothering about. The Minimum

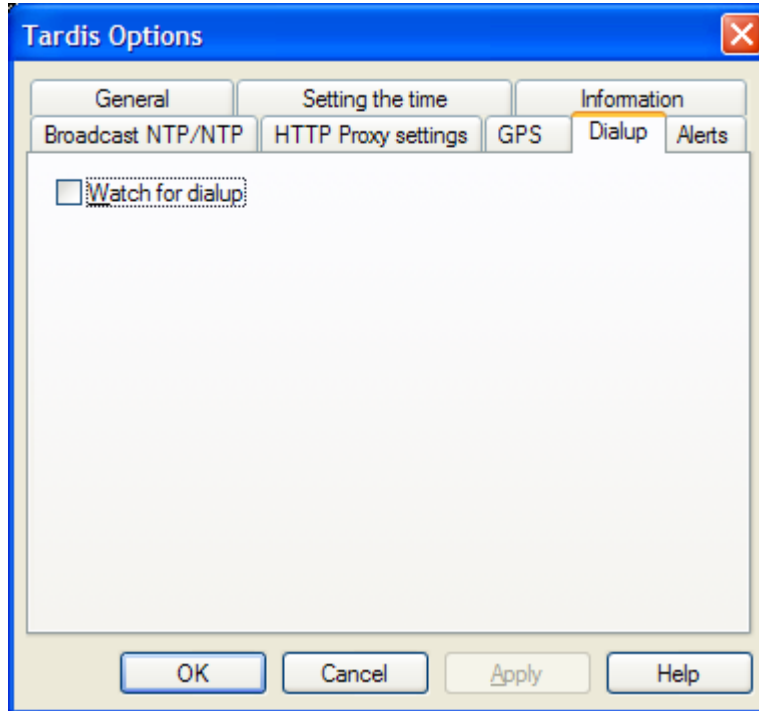


<b>correction allowed/Adjust clock frequency if below</b>	<p>correction allowed setting specifies what is reasonable. This is useful if you want to avoid frequent trivial changes to the time that might upset other applications. It is also possible to specify that any correction is allowed by sliding the control all the way to the left.</p> <p>On the <i>service</i> version this option's meaning changes if the 'Adjust clock frequency' is set. When this is set ALL time corrections are used even small ones. They are used to adjust the frequency of the PC's clock to make it drift less.</p> <p>When this is used the value of this setting determines when the clock is stepped rather than the frequency adjusted.</p>
<b>Maximum correction allowed</b>	<p>Tardis validates the time received from the server by checking that the amount of correction is not so far out that it must be wrong. The Allowable correction settings specify what is reasonable. This is useful if your timezone is wrong or your timeserver has gone mad.</p> <p>It is also possible to specify that any correction is allowed by sliding the control all the way to the left.</p>
<b>Allow correction of maximum size if larger</b>	<p>This option allows corrections up to but not larger than the maximum. If this is set and the maximum correction allowed is 1 second a correction of 10 seconds would result in a correction of 1 second. If the option weren't set the correction of 10 seconds would be rejected.</p>
<b>How often the time is set</b>	<p>This tells Tardis to get and set the time every so often. It depends on how bad your clock is. I use once every 60 minutes to keep mine in synch. Once a day may be enough for you.</p> <p>You can also tell Tardis to exit once it has corrected the time by sliding the control all the way to the left (Not on the <i>service</i> version). This is useful for setting the time when first starting the PC.</p>
<b>Adjust automatically</b>	<p><b>Note:</b> Once this has been set you may not be able to change the setting because Tardis will set the time and terminate before you get the chance. If this happens then hold down the SHIFT key and start Tardis. This will prevent it terminating.</p> <p>If this is set Tardis will gradually adjust how often it checks the time. This option works in conjunction with the Minimum allowed and Maximum allowed settings. If a correction is less than the minimum the time is increased. If a correction is made the time is decreased. The minimum correction setting effectively becomes the definition of the required accuracy.</p>
<b>Adjust clock frequency</b>	<p>This option is only present on the <i>service</i> version. When set Tardis will adjust the frequency of the PC's clock to keep it in synch rather than stepping the clock.</p>

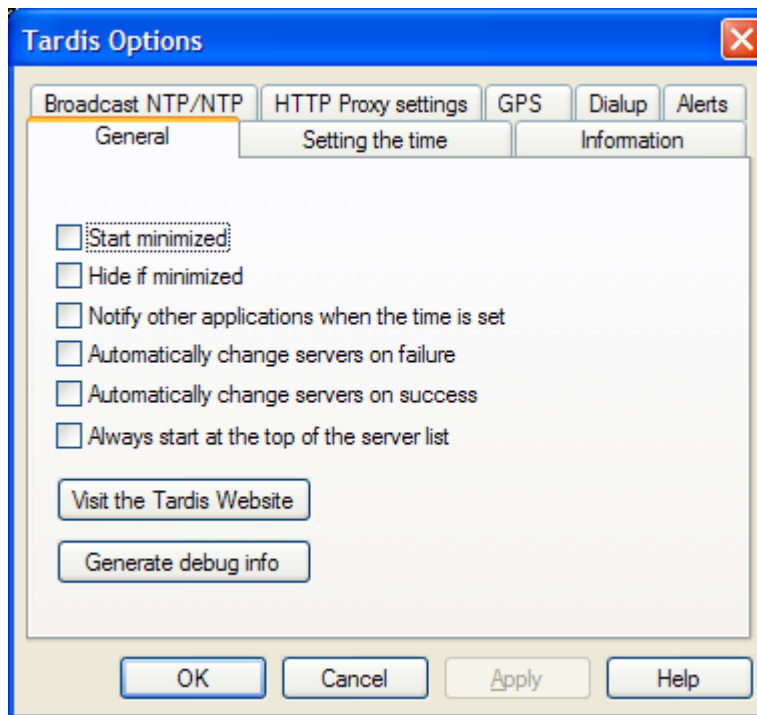
## 7.2 Dialup

Set this option to make Tardis watch for an active RAS dialup connection before doing anything. When the connection is established Tardis will check the time. If you don't have RAS installed this page will be absent. If you are using a dial on demand router rather than RAS this option won't work.

**Note:** Tardis won't initiate a connection for you and close it down automatically.



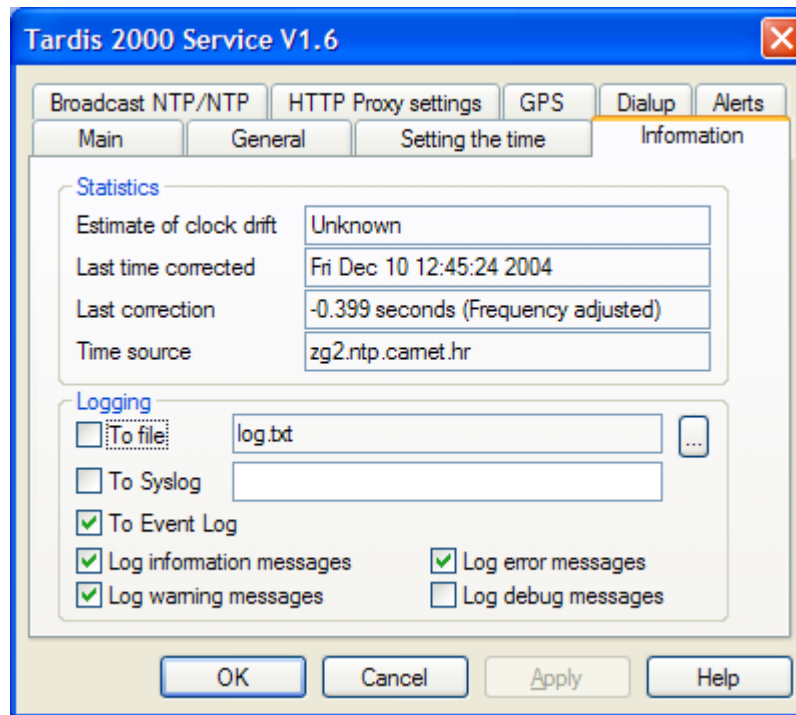
### 7.3 General options



<b>Start minimized</b>	If this is set then Tardis will start in a minimized state. (Not <i>service</i> version)
<b>Hide if minimized</b>	If this is set Tardis will hide itself if it is minimized. You can show it and hide it again by clicking on the Tardis Icon in the system tray. (Not <i>service</i> version)
<b>Notify other applications when the time is set</b>	Tardis can tell all the other running applications that the time has been changed. This may upset a small number of applications, e.g. Microsoft schedule, so it is off by default.
<b>Automatically change server on failure</b>	If this is set Tardis will automatically change to a different server if it cannot contact the one currently selected. It will cycle through all the servers in the list.
<b>Automatically change server on success</b>	If this is set Tardis will <i>automatically</i> change to a different server if it successfully contacts the one currently selected. It will cycle through all the servers in the list. This may be useful as a way of checking which servers are active.  <b>Note:</b> This option shouldn't be used in normal operation. It will cause the time to be checked every 2 seconds.
<b>Always start at the top of the server list</b>	If this is set Tardis will try to use the server at the top of the list first before moving on down the list until a good server is found. This differs from the normal behaviour where Tardis stays with the good server and never retries the servers that previously failed.
<b>Visit the Tardis Website</b>	If you press this button your web browser should start and you should be connected to the Tardis home page. If your web browser isn't set up quite right it won't work.
<b>Generate debug info</b>	If you have problems with Tardis press this button and include the output with any e-mail messages.

## 7.4 Information page

This page contains information on the last time correction and also controls what information is logged and where. The information is automatically updated as Tardis runs.



### 7.4.1 Statistics

#### Estimate of clock drift

This is an estimate of how much your PC's clock would drift without Tardis. It may require a long time (hours) before this value will settle down to a reasonable value. It is only going to settle down if you are using SNTP, Radio or GPS. The other protocols aren't accurate enough.

#### Last time corrected

When the time was last corrected.

#### Last correction

The size of the last correction.

#### Time source

Where the correction came from.

### 7.4.2 Logging

These options control what is logged. Messages may be placed in a file, sent to a Unix/Linux syslog server or saved to the system event log (Service version only). Tardis can use one or more of these destinations at the same time.

#### Log to file

If logging to a file press the ... button to set the location of the log file.

The file name may contain *strftime* escape sequences to allow the logging file to be named appropriately. E.g. %B will be replaced by the full month name so the log file %B.txt will be called April.txt in april and May.txt in may. This is the full list

%a	Abbreviated weekday name
%A	Full weekday name
%b	Abbreviated month name

%B	Full month name
%c	Date and time representation appropriate for locale
%d	Day of month as decimal number (01 – 31)
%H	Hour in 24-hour format (00 – 23)
%I	Hour in 12-hour format (01 – 12)
%j	Day of year as decimal number (001 – 366)
%m	Month as decimal number (01 – 12)
%M	Minute as decimal number (00 – 59)
%p	Current locale's A.M./P.M. indicator for 12-hour clock
%S	Second as decimal number (00 – 59)
%U	Week of year as decimal number, with Sunday as first day of week (00 – 53)
%w	Weekday as decimal number (0 – 6; Sunday is 0)
%W	Week of year as decimal number, with Monday as first day of week (00 – 53)
%x	Date representation for current locale
%X	Time representation for current locale
%y	Year without century, as decimal number (00 – 99)
%Y	Year with century, as decimal number
%z, %Z	Time-zone name or abbreviation; no characters if time zone is unknown
%%	Percent sign

### Log to a syslog server

Unix and Linux systems provide a networked equivalent to the Windows system event log. Tardis can send messages to one of these servers.

If syslog is selected you should enter the address of the syslog server. **Note:** the syslogd must be started with the `-r` option to enable it to receive remote messages from Tardis.

### Log to eventlog

The Tardis 2000 service logs to the system eventlog. This may be viewed using the Windows NT/2000/XP/2003 event log viewer. Tardis log messages are under the '*Application*' section.

**Log information messages** A minimal amount of logging will be recorded.

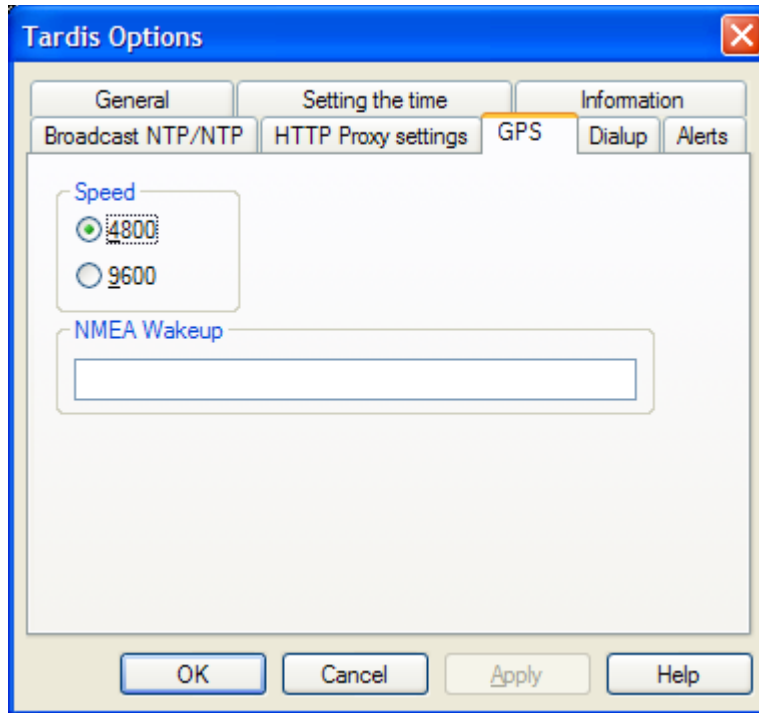
**Log warning messages** Warning messages will be logged.

**Log error messages** Errors will be logged

**Log debug messages** Debug messages are logged. If you have a problem with Tardis please switch this on and include the output in any e-mail that you may send me. It might help us find the problem.

## 7.5 GPS

This page configures the GPS connection.

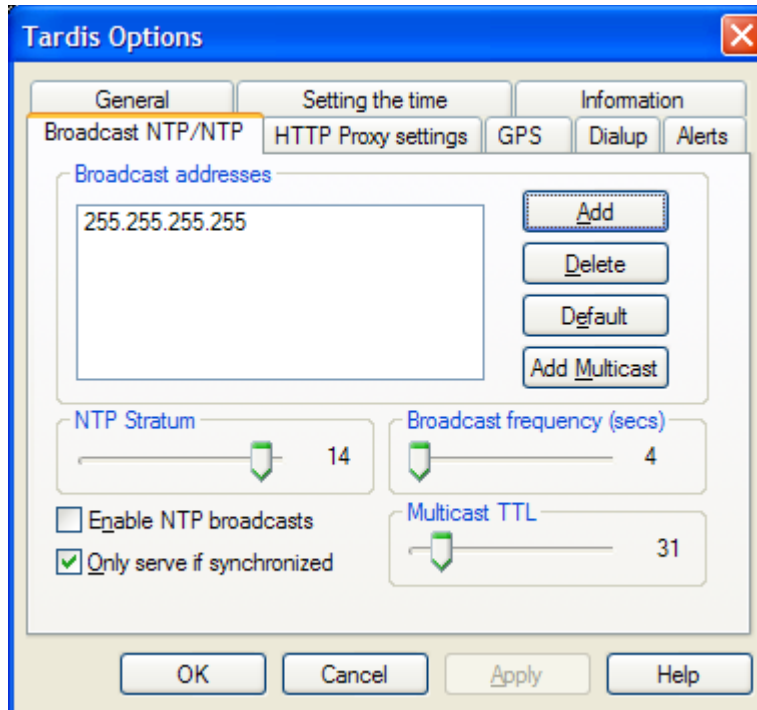


**Speed** Select the baudrate it uses.

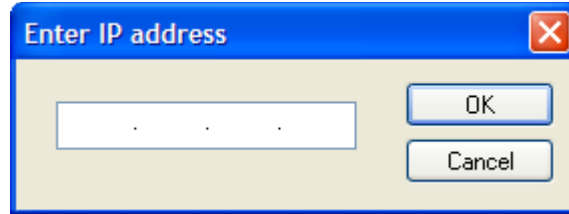
**NMEA Wakeup** If your GPS device requires a string to enable NMEA messages the NMEA wakeup string can be entered. The NMEA protocol is generally supported by most GPS systems. E.g. for the Delorme Tripmate the wakeup string is ASTRAL\r

## 7.6 Broadcast NTP/NTP

Tardis can broadcast NTP time information using broadcasts and multicasts. Use it if there are no NTP time broadcasts on your LAN. When the time is broadcast the *broadcast NTP* timeserver indicator will flash green (Not on *service* version). Tardis' companion program K9 is a minimal client for this protocol. See <http://www.mingham-smith.com/> for details of K9.



- Broadcast addresses** By default Tardis will broadcast to the local subnet broadcast address 255.255.255.255. Other addresses may be added and deleted to enable Tardis to broadcast to other subnets or even single PCs.
- Broadcast frequency** Use this setting to alter how often Tardis broadcasts the time.
- Enable NTP broadcasts** Select this to make Tardis broadcast NTP time information.
- Only serve if synchronized** Select this if you want Tardis to only broadcast and/or respond to NTP requests once it has synchronized the time. If a time synchronization alert happens the state will return to unsynchronized.
- NTP Stratum** This allows the NTP stratum to be set for all the NTP messages that Tardis sends. This affects the client and the server side of Tardis's NTP protocols. If using Tardis as a server for other clients it may be best to set the stratum to 1. This option is really for people who are familiar with NTP. Most people won't need to touch it.
- Add** This will enable you to add an IP address to which you want to broadcast NTP.



- Add multicast** Adds the default multicast address to the broadcast list.
- Multicast ttl** This sets the 'time to live' on the multicasts. It affects how far the multicasts travel from one subnet to another.
- Delete** Select this to delete the selected broadcast address.
- Default** Select this to set the broadcast address to the default.

### 7.6.1 Troubleshooting Broadcast NTP

One way to prove that things are mostly ok is to add the IP address of the target K9/Tardis machine into the list of broadcast addresses. That should prove that they can talk. The next thing to check is the broadcast address of the subnet. This can depend on the network.

If your subnet mask is 255.255.255.0 you can try 255.255.255.255 and 255.255.255.0 as the broadcast address (255.255.255.255 is the default)

You should also try the specific subnet broadcast addresses. If your subnet is 192.168.0.x try 192.168.0.255 and 192.168.0.0.

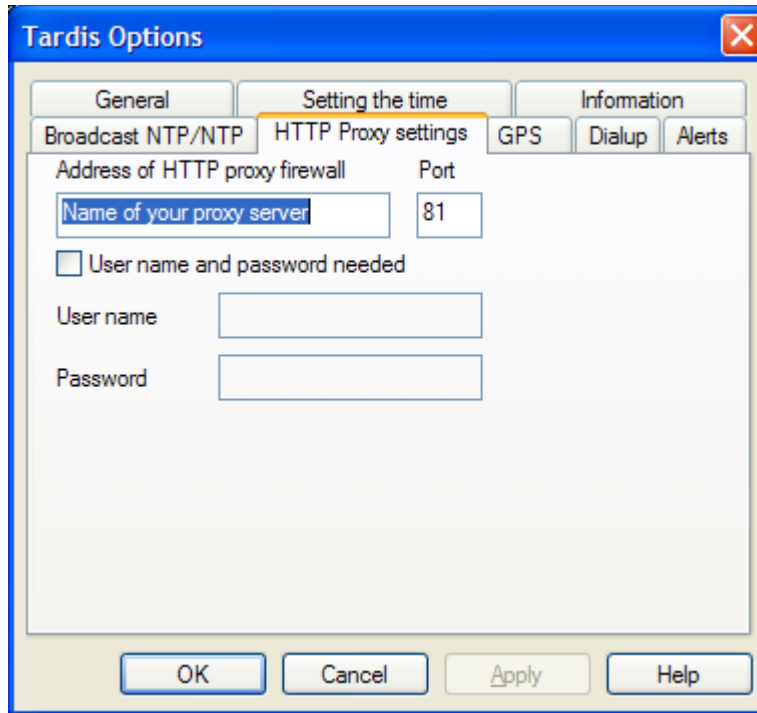
If your subnet mask differs from 255.255.255.0 on either or both machines let us know and I may be able to suggest other broadcast addresses to try.

Broadcasts are usually limited to local subnet. Multicasts are able to traverse multiple subnets if your network is set up to support it.



## 7.7 HTTP Proxy settings

Tardis may need to know the settings of your proxy server to work with the *http* protocol. The settings can probably be found in your web browser options. Tardis attempts to fill these in for you if a browser is configured.



### Address of HTTP proxy firewall

This is the name or IP address of your proxy/firewall server. You don't need to include a *http://* prefix.

### Port

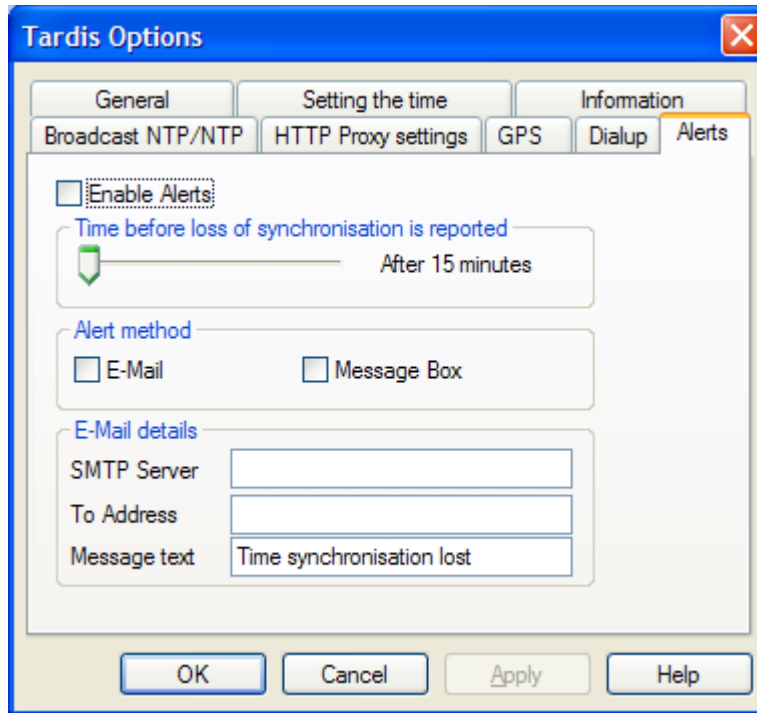
This is the port number that your proxy/firewall uses.

### User name and password needed

If your proxy/firewall requires authentication select this and enter your username and password.

## 7.8 Alerts

It is useful to know if Tardis has lost synchronization. This screen enables you to specify how long before an alert is raised.



### Enable Alerts

Alerts are normally off by default. Check this box to switch them on.

Set the time before an alert is raised. Note: make sure this is longer than the time between time checks otherwise it will always send an alert before the next time check.

When an alert happens the clock is marked as unsynchronized and NTP requests may not be served if the 'only server when synchronized' option is set.

### Alert method

Alerts may be sent as E-mail or a simple dialog box (Not service version). If neither is set Tardis will still react to the problem by changing server if it is configured to automatically switch servers on failure.

### E-mail details

SMTP Server	The server to send e-mail through
To address	The e-mail address of the recipient
Message text	The text of the message

## 8 Time Servers

Tardis provides servers for all the TCP/IP time protocols that it supports. I.e. you can run Tardis on your PC and other PC's will be able to retrieve the time from you. This enables you to lock PC's clocks together so that they always agree.

These are the things you need to do to use Tardis as a timeserver

Make sure that all the PCs are running TCP/IP.

If not then install TCP/IP from the settings->control panel->network->add->protocol dialog. Specify a subnet mask of '255.255.255.0' for the TCP/IP over the Ethernet adapters. This is for a class C address without subnetting. Then specify IP addresses beginning with 192.168.0.1 for your client computers.

Choose one machine as the server (i.e. the one that knows the correct time). Then specify the IP address of your Tardis server for the client machines under timeserver (as outlined in the documentation).

When a time service is accessed from another machine the appropriate indicator on the main Tardis screen will flash to indicate activity. (Not on the service version).

**Note:** Tardis always acts as a server for all the TCP/IP based protocols that it supports, no additional configuration is required.

## **9 Technical Support for Tardis 2000**

Support is available by Internet E-mail the address is [support@mingham-smith.com](mailto:support@mingham-smith.com)

The Tardis web site is [www.mingham-smith.com](http://www.mingham-smith.com)

## 10 Registering and Paying for Tardis 2000

The following page details the volume-based charges for Tardis 2000 and incorporates a registration form.

### 10.1 Payments

There are three ways to pay for registration:

1) **By cheque** payable to H.C. Mingham-Smith. Please post to the following address:

H. C. Mingham-Smith  
33 Arthur Rd.  
Wokingham,  
Berkshire RG41 2SS  
England.

2) **By Bank Transfer**

If you would prefer to pay by this method, please contact us on the following e-mail address to request bank account details.

E-mail address: [support@mingham-smith.com](mailto:support@mingham-smith.com)

3) **By Credit Card Online**

We have an arrangement with mycommerce.com who provide on-line credit card registration.

To register online, click on the relevant link

- [Tardis 2000](#)
- [Tardis 2000 NT Service](#)
- [Tardis 2000 Corporate License](#)
- [K9](#)

Please note that REGNOW cannot handle source code registrations.

### Invoices

If your company requires an invoice before sending payment, please e-mail us at [support@mingham-smith.com](mailto:support@mingham-smith.com) or post your purchase order to the above address.

**Charges** for registering your use of Tardis 2000 are based on the number of computers on which it is installed and are detailed on the registration forms which customers are requested to complete. Prices are quoted in US dollars, EUROs and £ Sterling. Customers outside the US or European Union are requested to convert the US dollar prices to the equivalent amount in their local currency.

Please note that **receipts** are normally sent via e-mail. If you require a receipt to be sent by post or a license to be issued, please request this when registering.

4) **Our Company Details**

HC Mingham-Smith Limited Registered in England No: 3676999.  
Registered Office: TSB House, 39A Peach Street, Wokingham, Berks RG40 1XJ  
VAT Registration Number: 642 4733 43

## **10.2 Upgrade Policy**

Customers who register their use of our software and pay the appropriate registration fee are entitled to a free upgrade to the next version of the software but are requested to pay an upgrade fee for the subsequent version.

The upgrade fee is 50% of the standard price.

### **Example**

Company X purchased 50 Tardis 2000 (v1.3) registrations in August 2001. They were entitled to a free upgrade to v1.4 but now wish to upgrade to v1.5.

The upgrade fee is

50% of 50 x US\$4.00 = US\$100

If, however, the company also wished to run the software on additional PCs they would be required to pay the standard fee for those registrations.

**Tardis 2000 Registration Form**  
**(For customers outside the European Union)**

UK VAT (Value Added Tax) does not apply to customers outside the European Union. The following prices are given in US\$. Non-US customers are invited to convert the following prices to their local currency.

**Quantity**

Please indicate the number of computers on which Tardis 2000 is installed and calculate the correct price

1 computer	_____	Computer at \$20 each =	_____
2 to 9 computers:	_____	computers at \$12 each =	_____
10 to 24 computers:	_____	computers at \$8 each =	_____
25 to 49 computers:	_____	computers at \$6 each =	_____
50 to 99 computers:	_____	computers at \$4 each =	_____
100 to 199 computers:	_____	computers at \$3 each =	_____
200+ computers:	_____	computers at \$2.50 each =	_____

**Corporate License**

Any number of copies for your whole company/organisation \$2000

**Source license**

The source of Tardis 2000 is available for \$200; please note that you are still required to pay the appropriate registration fee to run Tardis 2000 or software derived from the source.

Please provide the following information when registering:

Full Name/Name of company: \_\_\_\_\_

Your Address: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

E-Mail Address: \_\_\_\_\_

Windows Version: \_\_\_\_\_

Tardis 2000 version \_\_\_\_\_

Where did you obtain Tardis 2000? \_\_\_\_\_  
\_\_\_\_\_

Please send e-mail regarding Tardis 2000 to [support@mingham-smith.com](mailto:support@mingham-smith.com)

Visit the Tardis Home Page <http://www.mingham-smith.com/>

**Tardis 2000 Registration Form**

**(For customers in the European Union, but not in the UK)**

Customers in the European Union who use the software for business purposes are responsible for paying VAT at the appropriate rate in their home country. Customers registering their own personal use should pay VAT at the UK rate of 20% to HC Mingham-Smith Limited. The following prices are in EUROS and are exclusive of VAT. Prices may be converted to the customer's "home" currency if preferred.

**Quantity**

Please indicate the number of computers on which Tardis 2000 is installed and calculate the correct price

1 computer	_____	Computer at 23€ plus VAT each =	_____
2 to 9 computers:	_____	Computers at 14€ plus VAT each =	_____
10 to 24 computers:	_____	Computers at 9.50€ plus VAT each =	_____
25 to 49 computers:	_____	Computers at 7€ plus VAT each =	_____
50 to 99 computers:	_____	Computers at 5€ plus VAT each =	_____
100 to 199 computers:	_____	Computers at 3.50€ plus VAT each =	_____
200+ computers:	_____	Computers at 3€ plus VAT each =	_____

**Corporate License**

Any number of copies for your whole company/organisation 2300€ plus VAT

**Source license**

The source of Tardis 2000 is available for 230€ plus VAT; please note that you are still required to pay the appropriate registration fee to run Tardis 2000 or software derived from the source.

Please provide the following information when registering:

Full Name/Name of company: \_\_\_\_\_

Your Address: \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_

E-Mail Address: \_\_\_\_\_

Windows Version: \_\_\_\_\_

Tardis 2000 version \_\_\_\_\_

Where did you obtain Tardis 2000? \_\_\_\_\_  
 \_\_\_\_\_

Please send e-mail regarding Tardis 2000 to [support@mingham-smith.com](mailto:support@mingham-smith.com)

Visit the Tardis Home Page <http://www.mingham-smith.com/>



**Tardis 2000 Registration Form****(For UK customers)**

UK VAT (Value Added Tax) at 17.5% applies to sales to UK customers. The following prices are exclusive of VAT please add 17.5% to the final total.

**Quantity**

Please indicate the number of computers on which Tardis 2000 is installed and calculate the correct price

1 computer	_____	Computer at £12.50 plus VAT each =	_____
2 to 9 computers:	_____	Computers at £7.50 plus VAT each =	_____
10 to 24 computers:	_____	Computers at £5.00 plus VAT each =	_____
25 to 49 computers:	_____	Computers at £3.75 plus VAT each =	_____
50 to 99 computers:	_____	Computers at £2.50 plus VAT each =	_____
100 to 199 computers:	_____	Computers at £1.88 plus VAT each =	_____
200+ computers:	_____	Computers at £1.56 plus VAT each =	_____

**Corporate License**

Any number of copies for your whole company/organisation £1250 plus VAT

**Source license**

The source of Tardis 2000 is available for £125 plus VAT; please note that you are still required to pay the appropriate registration fee to run Tardis 2000 or software derived from the source.

Please provide the following information when registering:

Full Name/Name of company: \_\_\_\_\_

Your Address: \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_

E-Mail Address: \_\_\_\_\_

Windows Version: \_\_\_\_\_

Tardis 2000 version \_\_\_\_\_

Where did you obtain Tardis 2000? \_\_\_\_\_  
 \_\_\_\_\_

Please send e-mail regarding Tardis 2000 to [support@mingham-smith.com](mailto:support@mingham-smith.com)

Visit the Tardis Home Page <http://www.mingham-smith.com/>

## 11 Frequently asked questions

- Q** Tardis goes through the whole list of time servers without stopping  
**A** Make sure that you don't have the 'automatically change server on success' option set. This option is used the quickly scan through the server list. It shouldn't be used for normal operation.

- Q** I can't tell what the Tardis 2000 service is doing  
**A** Try running tardisnt from the command line so you can see what is going on in real time.

First , stop the service.

Then start a command prompt. Type

tardisnt debug

This will start the service in debug mode. It can be stopped with control-c.

- Q** I want to disable Tardis's time services  
**A** You can disable some of the Tardis time services but not the NTP service. It is too closely linked to the client side. From the start menu select 'run'. Type 'regedit' in the box and hit return.

You should see a thing rather like the explorer start.

Follow the folders down like you would in explorer to get to the  
 HKEY\_CURRENT\_USER\Software\Tardis\  
 folder

you should see a list of keys on the right side now.

Add keys as follows they are all of type DWORD.

```
UDPServer      0
TCPServer      0
867UDPServer   0
867TCPServer   0
```

Now shut down the regedit program and run tardis. All should be well.

This is for Tardis 2000 for 95/98/ME. The same applies to Tardis 2000 NT Service but the keys are in HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tardis\Parameters\

- Q** Tardis 2000 may give error 87 messages after being able to sync one time.  
**A** This can be fixed as follows.

Win95 comes with Winsock 1, Tardis2000 requires Winsock 2, which can be downloaded free from Microsoft's Web site.

To get Winsock 2, go to Microsoft's Windows Update web page (a menu item in IE 5.5's "tools" menu) and do a search for Winsock 2.

Microsoft recommends that you also install a Y2K fix after installing either Winsock 2 or DUN 1.3.

The link for that is right there on the Winsock 2 page, as are the instructions for downloading & installing.

- Q** Tardis starts then immediately disappears  
**A** The 'How often the time is set' setting can be set to tell Tardis to exit once it has corrected the time by sliding the control all the way to the left.

Once this has been set you may not be able to change the setting because Tardis will set the time and terminate before you get the chance. If this happens then hold down the SHIFT key then start Tardis. This will prevent it terminating.

- Q** What is the format of the import/export file  
**A** The import/export file format is pretty simple.  
 One line per server.  
 The fields are separated by | characters.

field 1 is the name  
 field 2 is the address  
 field 3 is the protocol (0-9)  
 field 4 is whether a proxy server is used (for HTTP)  
 field 5 is whether unsynchronized NTP should be rejected (SNTP)

e.g.  
 ntp.demon.co.uk|ntp.demon.co.uk|0|0|1

- Q** Tardis can't get the time from internet time servers  
**A** Firewalls or proxy servers will block most of the time protocols that Tardis uses.  
 The one that might work is the HTTP protocol.

Another possibility is to open your firewall/proxy tcp port 37 and use the RFC868/tcp protocol.

- Q** Tardis doesn't send from port udp/123, which means my firewall doesn't allow the responses through/generates bogus virus warnings.  
**A** By default Tardis SENDs to port 123 but the SOURCE port it uses is allocated by the system so the responses come in on a different port number. There is a tweak in the registry you can make to force Tardis to lock the source port to 123.

the key to add is  
 HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tardis\Parameters\SNTPFix  
 it is a DWORD and should be set to 1

The same applies to Tardis 2000 but the key is  
 HKEY\_CURRENT\_USER\Software\Tardis\SNTPFix

- Q** I want Tardis to set the time at an exact time of day.  
**A** The best way to do this is to use the non-service version of Tardis and run it so that it terminates after getting the time. You then run Tardis at the time you want using the NT AT command or win 95/98's task scheduler.

- Q** Tardis doesn't synchronize with my GPS/Radio clock.  
**A** Tardis may not synchronize with a GPS source for various reasons.

Tardis may not see the NMEA messages because the wrong serial port is being used or the baud rate is incorrect.

If it is seeing messages but isn't synchronizing it is probably because the GPS device hasn't had

sufficient time to see enough satellites to synchronize.

Before Tardis 2000 V1.3 the GPS device was required to send a message every second. Some GPS devices don't do this.

**Q** Where are the license keys?

**A** The reason we don't have license keys is as follows.

Some of the most annoying things about shareware are the crippled and/or time limited demos that are normally available.

As soon as I get one of these I usually don't get much further, I don't try the program and don't end up buying it.

If we did have license keys there would be the problems of crackers creating their own, having to issue new ones for users who lost theirs. Tracking how many have been issued etc. etc.

We went for the friendliest, lowest cost way of doing things.

The downside is that fewer people pay. We will live with that to keep everyone's life simple. Anyway, most of the registrations come from companies that regularly audit their software to check if it has been paid for so our major customers sort of police themselves.

The opposite extreme to this kind of scheme is something like Windows XP's licensing. I can't begin to tell you how popular THAT is...

**Q** Where does the NT service log messages

**A** Tardis 2000 NT Service logs events to the system event log.

Use the event log viewer to view them.

Tardis logs messages under the 'application' section.

To control logging select the 'information' option page on the Tardis control panel and switch on or off the amount of logging you require.

**Q** My GPS time is a little off.

**A** The problem comes from trying to get a GPS device to TELL you the accurate time.

NMEA supports several sentences that contain the time.

Tardis supports all of these and will use the best one it sees. \$GPZDA is best because it is designed to tell the time.

The next best is \$GPRMC. This tells you the time of the last position fix (which is usually in the last few seconds).

If your GPS only supports \$GPRMC you should find that the offset is fairly constant. You can manually tell Tardis this offset as follows.

from the start menu select 'run'. Type 'regedit' in the box and hit return.

You should see a thing rather like the explorer start.

Follow the folders down like you would in explorer to get to the  
HKEY\_CURRENT\_USER\Software\Tardis\  
folder

you should see a list of keys on the right side now.

Add a key called NMEAAdjust of type 'String'. This is the NEMA time offset in milliseconds.

Now shut down the regedit program and run tardis. All should be well.

This is for Tardis 95/98. The same applies to TardisNT but the key is

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tardis\Parameters\NMEAAdjust

**Q** Tardis gets but doesn't set the time.

**A** The problem might be the account that the tardis service is running under. When you run it under debug you are running it as 'you' and 'you' probably have administrator privileges. When it runs as a service it runs under the local SYSTEM account by default.

Use the services control panel to change this to 'administrator' and it should start working.

**Q** How does the service version differ from the application version?

**A** The service version is a windows service that has a couple of differences to the application version.

Tardis is split into 2 parts, tardisnt.exe which is the service that actually does the time synching and tardis.cpl that is the control panel that presents the user interface. These 2 parts aren't connected in any way except by the settings in the registry. The control panel sets up the settings and the service uses them.

Tardisnt.exe is running so the service is install and running, you can use the control panel to start and stop the service but you won't see anything happening as such.

If you want all the normal feedback and the icon in the system tray etc. you CAN run the Application version of Tardis too.

To check what the service is doing you can look in the eventlog under the application section or run it under debug mode from a command line prompt like so.

tardisnt debug

Remember to stop the service first before running it as debug.

**Q** Can I use w32time and Tardis on the same PC?

**A** Tardis and w32time will not coexist on the same PC because they are both trying to do the same job using the same resources. In this case the resource is the udp port 123.

Actually, Tardis doesn't mind if w32time is running too much but w32time doesn't like Tardis using the same resources. If you can arrange w32time to start first you can just about get away with running them both.

Tardis basically does the same job as w32time (but better obviously). Why would you want to use w32time? Tardis is NTP compliant too.

**Q.** Why are there 2 versions?

**A.** The Windows NT/2000/XP/2003 version (Tardis 2000 Service) runs as a service so it is still working when nobody is logged in, important for Server machines that may be logged off but are being used as

servers. The other version is for Windows 95/98//ME/2000/NT/XP/2003 where Tardis is run as a normal application.

**Q.** Who is Tardis 2000 for?

**A.** Tardis is designed to be used by almost anyone. It is for dialup users and LAN users.

**Q.** Why is it called Tardis 2000?

**A.** Tardis is a time machine that appeared on BBC TV's Dr. Who program. The 2000 has been added, not to indicate Year 2000 compliance (Tardis has always been compliant), but to indicate that it is a complete rewrite i.e. not just the next version of Tardis95/NT.

**Q.** Where do I find the latest version of Tardis?

**A.** Visit the Tardis home page <http://www.mingham-smith.com>

**Q.** What is NTP?

**A.** NTP is an Internet protocol for time synchronization. For details on it look at <http://www.ntp.org> on the World Wide Web.

**Q.** What is broadcast NTP?

**A.** NTP includes an option to broadcast a timesignal.

**Q.** My LAN has no broadcast NTP, how do I set it up?

**A.** To do the job properly you will need to get a copy of XNTP, this is an implementation of NTP that supports many machine types (mostly Unix machines although there is a port to NT). See the NTP home web site <http://www.ntp.org>

Tardis will also run in a time server mode where it will broadcast the time from the PC on which it is run.

**Q.** Why are only some of the PCs being updated when I use broadcasts?

**A.** Time broadcasts are limited to the local subnet, make sure that Tardis is set up to broadcast to all the required subnets or use multicasts.

**Q.** Year 2000 compliance

**A.** Tardis is and always has been Y2K compliant. It does rely on the underlying operating system for some services and the hardware clock of course so if there is a problem with Windows or the PC's bios then Tardis may not be able to correct it. Tardis does not hold dates in yy/mm/dd form internally so Y2K should not be an issue. Tardis uses Unix standard time representations internally (i.e. time is held as number of seconds since 1970). I recommend that you verify for yourself that Tardis is Y2K compliant as a final check that it is suitable for your purposes.

**Q.** Can I use Tardis with ntp/xntp?

- A.** Yes, you can use Tardis as a server for xntp clients or visa versa.
- Q.** How do I set up Tardis as a server?
- A.** Tardis can both act as a server for other time clients. Tardis is set up to always act as a server. You don't need to do anything but run it. Both clients and server machines need to have TCP/IP installed. On client machines set the hostname or IP address ( something like 123.45.67.89) of the server as the source of time information. You can even set Tardis to use itself as a server by setting the time source as localhost or 127.0.0.1. Not very useful but it does demonstrate the technique. When the client accesses the server the relevant service indicator on the server's dialog should blink RED to show that the server is being accessed. (Not on the Service version)
- Q.** Can I use Tardis when I have a firewall with a proxy server in the way?
- A.** Yes, use the HTTP protocol.
- Q.** Does Tardis support Mac, OS/2, Novell, DOS, etc.
- A.** No.
- Q.** Tardis doesn't seem to handle daylight/summer time in Europe?
- A.** The windows Date/Time control panel must be set to the correct timezone for your machine for Tardis to work properly. Make sure that this is correct. If Tardis is still setting the time wrongly it could be that Windows 95/98/ME's rules are wrong for your timezone. The following Windows 95/98/ME timezone rules are known to be wrong.
- Some parts of Australia
- Russia
- Europe (rules changed in 1996)
- To fix the rules get hold of the Microsoft kernel toys and use the timezone editor to correct them.
- Q.** Where can I get NTP for UNIX?
- A.** <http://www.ntp.org>
- Q.** The NT/2000/XP/2003 service version doesn't update the clock when nobody is logged in?
- A.** Use the setting->control panel->services to edit Tardis's startup settings. Set the Tardis service to run under the 'Administrator' account

## 12 Time Protocols

Tardis can find out the correct time in many different ways. The following sections describe the different protocols that Tardis supports in some detail. Some of the descriptions are quite technical but don't get worried, you don't need to understand them to use Tardis.

### 12.1 NTP Broadcast protocol

If you have an NTP server on your LAN it may be configured to broadcast time information. Set Tardis to listen for these broadcasts with the NTP broadcast protocol. This is the lowest possible use of your network bandwidth, Tardis does not need to be told the name of the server and it does not need to ask for the time. If the name *is* specified then Tardis will only listen to broadcasts from that address.

Many Tardis clients can be set with a single broadcast. Tardis' companion program K9 is a minimal client for this protocol. If you are only using NTP broadcasting K9 is a better choice than Tardis because it is easier to install and configure.

### 12.2 NMEA

GPS cards generally support the National Marine Electronics Association (NMEA) protocol. Tardis can use this protocol to determine the time. Tardis can use the following messages defined in the protocol.

- \$GPZDA
- \$GPRMC
- \$GPGGA

If Tardis detects that more than one of these messages is present it will use the 'best'. \$GPZDA is considered the preferred message to use, followed by \$GPRMC then \$GPGGA.

An example of these types of message is shown below.

```
$GPRMC,235956,V,2956.8135,N,09353.2197,W,0.000,0.0,120895,4.2,E*7A
```

The message contains the date and time amongst other things.

### 12.3 HTTP Protocol

Tardis is sometimes used in situations where it does not have access to the Internet e.g. on a company's LAN. Users may have access to the Internet using a web browser but they aren't allowed other protocols, including the time related ones. If no other sources of time are available on the company Intranet the only way Tardis can find out the time is by pretending to be a web browser.

The HTTP protocol allows you to do this. Part of the HTTP protocol allows Tardis to find out the time even when working through a firewall or a proxy server.

Enter the address of a web site that appears to know the correct time. Many of the bigger sites seem to have good time keeping, <http://www.altavista.digital.com> for example. Tardis will pretend to be a browser and retrieve the time. Remember to set up the proxy settings if one is used.

The time retrieved is generally not nearly as accurate as the other time protocols. It could easily be a several seconds out because of network delays and proxy server slowness.



## 12.4 Galleon Radio Clock

### Atomic Radio Clocks give stratum 1 time source

Synchronise your computers time and date to the most accurate clock in the world.

Using Atomic Radio Clocks the time received is always correct ensuring that your computer is synchronised to the world standard.

Galleon Atomic Radio Clocks gives you a **Stratum 1** time source so you have total confidence it's correct.

- Over 16,000 Galleon atomic radio clocks installed
- Server time is always correct.
- Help combat fraud by providing accurate auditable time stamping
- Log exactly when events occur
- A single clock will synchronise the time on a complete network of computers.

### Atomic Radio Clocks are available for

- **America, the WWVB** time signal transmitter.
- **Europe the German DCF** time signal transmitter.
- **GMT the British MSF** time signal transmitter.

Using the national time transmitter provides the strongest time signal, therefore able to receive time signal across the country.

For specific product details

<http://www.AtomicRadioClocks.com> or <http://www.timesynch.co.uk>

<mailto:sales@galleon-uk.com>

## 12.5 Kallisto GPS Card

### Low Cost GPS time reference available in short ISA AT card format

*Harness the precision time provided by the Global Positioning Service (GPS) satellites within your computer. This low cost unit allows you to synchronise your equipment or PC system clock to microsecond accuracy - banish those drifting clocks forever!*

#### Key Features of the Kallisto GPS Card

16 bit half length AT bus card (8 data bits used but all AT IRQs available)

Based on Rockwell 12 channel Jupiter GPS module

Provides GPS position and time

1 pulse per second signal available as hardware interrupt and external output. Synchronised to second boundary, accurate to one microsecond.

External 1 pulse per second output qualified by software mask to allow triggering on specified second time mark.

Two uncommitted TTL outputs.

10kHz frequency standard available as external output

Requires only a simple external passive or active GPS antenna

Battery back up of internal time of day clock to speed up acquisition of GPS time on power up

All specifications and accuracies subject to change.

#### Method of Operation

The Rockwell GPS module presents all of its output information as a series of packets over a serial data link. The data can be presented in either Rockwell binary or industry standard NMEA ASCII format. The AT card contains a UART to decode the serial stream and make it available to the PC bus. The UART design is functionally identical to a standard PC serial port so existing serial driver software can be used. Every second (binary mode) the GPS module generates a packet of 108 bytes. Amongst other things this packet contains the current position and UTC time. The UTC time can be used to set the computer's operating system clock. To obtain more precise timing information the module's 1 pulse per second (1pps) signal is used. This signal is synchronised to the second boundary and is connected to an IRQ and is also made available as an external output. Clock driver software can be written to firstly obtain the initial time via the serial link and set the operating system clock. The interrupt service routine for the 1pps signal simply increments the time by one second and then sets the operating system clock to the new time. Note: if this method is used the card will use 2 IRQs - one for the serial port and one for the 1pps interrupt.

#### Qualified 1 pulse per second (PPS) output

A unique feature of this card is the qualified 1pps external output. the 1pps signal can be masked off with a bit in a software register thus allowing external trigger signals to be generated at a specified time whilst retaining synchronisation with GPS time.

#### Software

The board is supplied with a Windows application to set the operating system clock. Programming is fairly simple and full information is provided in the Rockwell manual that is supplied with the card. A DOS test program provided by Rockwell, together with the source code, is also available. A QNX driver is also available on request.

#### Antennas

The antenna must have a clear unobstructed view of the sky. If the feeder length is less than about 2 metres a low cost passive antenna will be adequate. For feeders longer than 2 metres an active antenna should be used. Suitable antennas and cabling can be supplied if required.

#### Applications include

Synchronisation of equipment

Master time reference for networked computer systems

Accurate time stamp for data loggers and similar applications

Positional information to 100m accuracy

**For additional information and pricing contact:**

Greening Technology  
30 Prospect Road  
Kibworth Beauchamp  
Leicester  
LE8 0HX  
UK

Tel: 0116 279 6500 Intl. +44-116 -279-6500 Fax: 0116 279 6501 Intl. +44-116 -279-6501

email regarding Kallisto should be sent to: [jth@ion.le.ac.uk](mailto:jth@ion.le.ac.uk)

This will be forwarded to Greening Technology. <http://ion.le.ac.uk/kallisto/kallisto.html>

## **12.6 RFC2030 (SNTP)**

Network Working Group D. Mills

Request for Comments: 2030 University of Delaware

Obsoletes: 1769 October 1996

Category: Informational

### **Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI**

#### **12.6.1 Status of this Memo**

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

#### **12.6.2 Abstract**

This memorandum describes the Simple Network Time Protocol (SNTP) Version 4, which is an adaptation of the Network Time Protocol (NTP) used to synchronize computer clocks in the Internet. SNTP can be used when the ultimate performance of the full NTP implementation described in RFC-1305 is not needed or justified. When operating with current and previous NTP and SNTP versions, SNTP Version 4 involves no changes to the NTP specification or known implementations, but rather a clarification of certain design features of NTP which allow operation in a simple, stateless remote-procedure call (RPC) mode with accuracy and reliability expectations similar to the UDP/TIME protocol described in RFC-868.

The only significant protocol change in SNTP Version 4 over previous versions of NTP and SNTP is a modified header interpretation to accommodate Internet Protocol Version 6 (IPv6) [DEE96] and OSI [COL94] addressing. However, SNTP Version 4 includes certain optional extensions to the basic Version 3 model, including an anycast mode and an authentication scheme designed specifically for multicast and anycast modes. While the anycast mode extension is described in this document, the authentication scheme extension will be described in another document to be published later. Until such time that a definitive specification is published, these extensions should be considered provisional.

This memorandum obsoletes RFC-1769, which describes SNTP Version 3. Its purpose is to correct certain inconsistencies in the previous document and to clarify header formats and protocol operations for current NTP Version 3 (IPv4) and proposed NTP Version 4 (IPv6 and OSI), which are also used for SNTP. A working knowledge of the NTP Version 3 specification RFC-1305 is not required for an implementation of SNTP.

#### **12.6.3 Introduction**

The Network Time Protocol (NTP) Version 3 specified in RFC-1305 [MIL92] is widely used to synchronize computer clocks in the global Internet. It provides comprehensive mechanisms to access national time and frequency dissemination services, organize the time-synchronization subnet and adjust the local clock in each participating subnet peer. In most places of the Internet of today, NTP provides accuracies of 1-50 ms, depending on the characteristics of the synchronization source and network paths.

RFC-1305 specifies the NTP Version 3 protocol machine in terms of events, states, transition functions and actions and, in addition, engineered algorithms to improve the timekeeping quality and mitigate among several synchronization sources, some of which may be faulty. To achieve accuracies in the low milliseconds over paths spanning major portions of the Internet of today, these intricate algorithms, or their functional equivalents, are necessary. However, in many cases accuracies in the order of significant fractions of a second are acceptable. In such cases, simpler protocols such as the Time Protocol [POS83], have been used for this purpose. These protocols usually involve an RPC exchange where the client requests the time of day and the server returns it in seconds past some known reference epoch.

NTP is designed for use by clients and servers with a wide range of capabilities and over a wide range of network delays and jitter characteristics. Most users of the Internet NTP synchronization subnet of today use a software package including the full suite of NTP options and algorithms, which are relatively complex, real-time applications (see <http://www.eecis.udel.edu/~ntp>). While the software has been ported to a wide variety of hardware platforms ranging from personal computers to supercomputers, its sheer size and complexity is not appropriate for many applications. Accordingly, it is useful to explore alternative access strategies using simpler software appropriate for less stringent accuracy expectations.

This document describes the Simple Network Time Protocol (SNTP) Version 4, which is a simplified access strategy for servers and clients using NTP Version 3 as now specified and deployed in the Internet, as well as NTP Version 4 now under development. The access paradigm is identical to the UDP/TIME Protocol and, in fact, it should be easily possible to adapt a UDP/TIME client implementation, say for a personal computer, to operate using SNTP. Moreover, SNTP is also designed to operate in a dedicated server configuration including an integrated radio clock. With careful design and control of the various latencies in the system, which is practical in a dedicated design, it is possible to deliver time accurate to the order of microseconds.

SNTP Version 4 is designed to coexist with existing NTP and SNTP Version 3 clients and servers, as well as proposed Version 4 clients and servers. When operating with current and previous versions of NTP and SNTP, SNTP Version 4 requires no changes to the protocol or implementations now running or likely to be implemented specifically for NTP or SNTP Version 4. To a NTP or SNTP server, NTP and SNTP clients are indistinguishable; to a NTP or SNTP client, NTP and SNTP servers are indistinguishable. Like NTP servers operating in non-symmetric modes, SNTP servers are stateless and can support large numbers of clients; however, unlike most NTP clients, SNTP clients normally operate with only a single server. NTP and SNTP Version 3 servers can operate in unicast and multicast modes. In addition, SNTP Version 4 clients and servers can implement extensions to operate in anycast mode.

It is strongly recommended that SNTP be used only at the extremities of the synchronization subnet. SNTP clients should operate only at the leaves (highest stratum) of the subnet and in configurations where no NTP or SNTP client is dependent on another SNTP client for synchronization. SNTP servers should operate only at the root (stratum 1) of the subnet and then only in configurations where no other source of synchronization other than a reliable radio or modem time service is available. The full degree of reliability ordinarily expected of primary servers is possible only using the redundant sources, diverse subnet paths and crafted algorithms of a full NTP implementation. This extends to the primary source of synchronization itself in the form of multiple radio or modem sources and backup paths to other primary servers should all sources fail or the majority deliver incorrect time. Therefore, the use of SNTP rather than NTP in primary servers should be carefully considered.

An important provision in this document is the reinterpretation of certain NTP Version 4 header fields which provide for IPv6 and OSI addressing and optional anycast extensions designed specifically for multicast service. These additions are in conjunction with the proposed NTP Version 4 specification, which will appear as a separate document. The only difference between the current NTP Version 3 and proposed NTP Version 4 header formats is the interpretation of the four-octet Reference Identifier field, which is used primarily to detect and avoid synchronization loops. In Version 3 and Version 4 primary (stratum-1) servers, this field contains the four-character ASCII reference identifier defined later in this document. In Version 3 secondary servers and clients, it contains the 32-bit IPv4 address of the synchronization source. In Version 4 secondary servers and clients, it contains the low order 32 bits of the last transmit timestamp received from the synchronization source.

In the case of OSI, the Connectionless Transport Service (CLTS) is used [ISO86]. Each SNTP packet is transmitted as the TS-Userdata parameter of a T-UNITDATA Request primitive. Alternately, the header can be encapsulated in a TPDU which itself is transported using UDP [DOB91]. It is not advised that NTP be operated at the upper layers of the OSI stack, such as might be inferred from [FUR94], as this could seriously degrade accuracy. With the header formats defined in this document, it is in principle possible to interwork between servers and clients of one protocol family and another, although the practical difficulties may make this inadvisable.

In the following, indented paragraphs such as this one contain information not required by the formal protocol specification, but considered good practice in protocol implementations.

#### 12.6.4 Operating Modes and Addressing

SNTP Version 4 can operate in either unicast (point to point), multicast (point to multipoint) or anycast (multipoint to point) modes. A unicast client sends a request to a designated server at its unicast address and expects a reply from which it can determine the time and, optionally, the roundtrip delay and local clock offset relative to the server. A multicast server periodically sends a unsolicited message to a designated IPv4 or IPv6 local broadcast address or multicast group address and ordinarily expects no requests from clients. A multicast client listens on this address and ordinarily sends no requests. An anycast client sends a request to a designated IPv4 or IPv6 local broadcast address or multicast group address. One or more anycast servers reply with their individual unicast addresses. The client binds to the first one received, then continues operation in unicast mode.

Multicast servers should respond to client unicast requests, as well as send unsolicited multicast messages. Multicast clients may send unicast requests in order to determine the network propagation delay between the server and client and then continue operation in multicast mode.

In unicast mode, the client and server end-system addresses are assigned following the usual IPv4, IPv6 or OSI conventions. In multicast mode, the server uses a designated local broadcast address or multicast group address. An IP local broadcast address has scope limited to a single IP subnet, since routers do not propagate IP broadcast datagrams. On the other hand, an IP multicast group address has scope extending to potentially the entire Internet. The scoping, routing and group membership procedures are determined by considerations beyond the scope of this document. For IPv4, the IANA has assigned the multicast group address 224.0.1.1 for NTP, which is used both by multicast servers and anycast clients. NTP multicast addresses for IPv6 and OSI have yet to be determined.

Multicast clients listen on the designated local broadcast address or multicast group address. In the case of local broadcast addresses, no further provisions are necessary. In the case of IP multicast addresses, the multicast client and anycast server must implement the Internet Group Management Protocol (IGMP) [DEE89], in order that the local router joins the multicast group and relays messages to the IPv4 or IPv6 multicast group addresses assigned by the IANA. Other than the IP addressing conventions and IGMP, there is no difference in server or client operations with either the local broadcast address or multicast group address.

It is important to adjust the time-to-live (TTL) field in the IP header of multicast messages to a reasonable value, in order to limit the network resources used by this (and any other) multicast service. Only multicast clients in scope will receive multicast server messages. Only co-operating anycast servers in scope will reply to a client request. The engineering principles which determine the proper value to be used are beyond the scope of this document.

Anycast mode is designed for use with a set of co-operating servers whose addresses are not known beforehand by the client. An anycast client sends a request to the designated local broadcast or multicast group address as described below. For this purpose, the NTP multicast group address assigned by the IANA is used. One or more anycast servers listen on the designated local broadcast address or multicast group address. Each anycast server, upon receiving a request, sends a unicast reply message to the originating client. The client then binds to the first such message received and continues operation in unicast mode. Subsequent replies from other anycast servers are ignored.

In the case of SNTP as specified herein, there is a very real vulnerability that SNTP multicast clients can be disrupted by misbehaving or hostile SNTP or NTP multicast servers elsewhere in the Internet, since at present all such servers use the same IPv4 multicast group address assigned by the IANA. Where necessary, access control based on the server source address can be used to select only the designated server known to and trusted by the client. The use of cryptographic authentication scheme defined in RFC-1305 is optional; however, implementers should be advised that extensions to this scheme are planned specifically for NTP multicast and anycast modes.

While not integral to the SNTP specification, it is intended that IP broadcast addresses will be used primarily in IP subnets and LAN segments including a fully functional NTP server with a number of dependent SNTP multicast clients on the same subnet, while IP multicast group addresses will be used only in cases where the TTL is engineered specifically for each service domain.

In NTP Version 3, the reference identifier was often used to walk-back the synchronization subnet to the root (primary server) for management purposes. In NTP Version 4, this feature is not available, since the addresses are longer than 32 bits. However, the intent in the protocol design was to provide a way to detect and avoid loops. A peer could determine that a loop was possible by comparing the contents of this field with the IPv4 destination address in the same packet. A NTP Version 4 server can accomplish the same thing by comparing the contents of this field with the low order 32 bits of the originate timestamp in the same packet. There is a small possibility of false alarm in this scheme, but the false alarm rate can be minimized by randomizing the low order unused bits of the transmit timestamp.

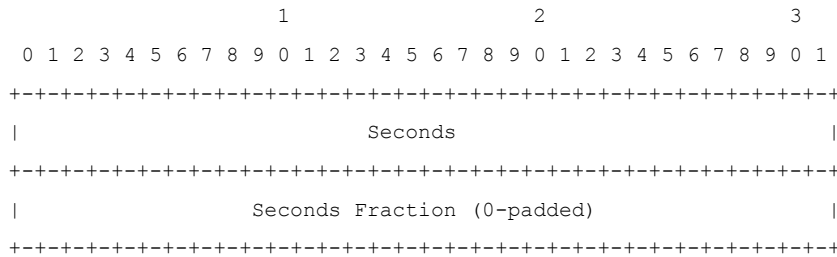
### 12.6.5 NTP Timestamp Format

SNTP uses the standard NTP timestamp format described in RFC-1305 and previous versions of that document. In conformance with standard Internet practice, NTP data are specified as integer or fixed-point quantities, with bits numbered in big-endian fashion from 0 starting at the left, or high-order, position. Unless specified otherwise, all quantities are unsigned and may occupy the full field width with an implied 0 preceding bit 0.

Since NTP timestamps are cherished data and, in fact, represent the main product of the protocol, a special timestamp format has been established. NTP timestamps are represented as a 64-bit unsigned fixed-point number, in seconds relative to 0h on 1 January 1900. The integer part is in the first 32 bits and the fraction part in the last 32 bits. In the fraction part, the non-significant low order can be set to 0.

It is advisable to fill the non-significant low order bits of the timestamp with a random, unbiased bitstring, both to avoid systematic roundoff errors and as a means of loop detection and replay detection (see below). One way of doing this is to generate a random bitstring in a 64-bit word, then perform an arithmetic right shift a number of bits equal to the number of significant bits of the timestamp, then add the result to the original timestamp.

This format allows convenient multiple-precision arithmetic and conversion to UDP/TIME representation (seconds), but does complicate the conversion to ICMP Timestamp message representation, which is in milliseconds. The maximum number that can be represented is 4,294,967,295 seconds with a precision of about 200 picoseconds, which should be adequate for even the most exotic requirements.



Note that, since some time in 1968 (second 2,147,483,648) the most significant bit (bit 0 of the integer part) has been set and that the 64-bit field will overflow some time in 2036 (second 4,294,967,296). Should NTP or SNTP be in use in 2036, some external means will be necessary to qualify time relative to 1900 and time relative to 2036 (and other multiples of 136 years). There will exist a 200-picosecond interval, henceforth ignored, every 136 years when the 64-bit field will be 0, which by convention is interpreted as an invalid or unavailable timestamp.

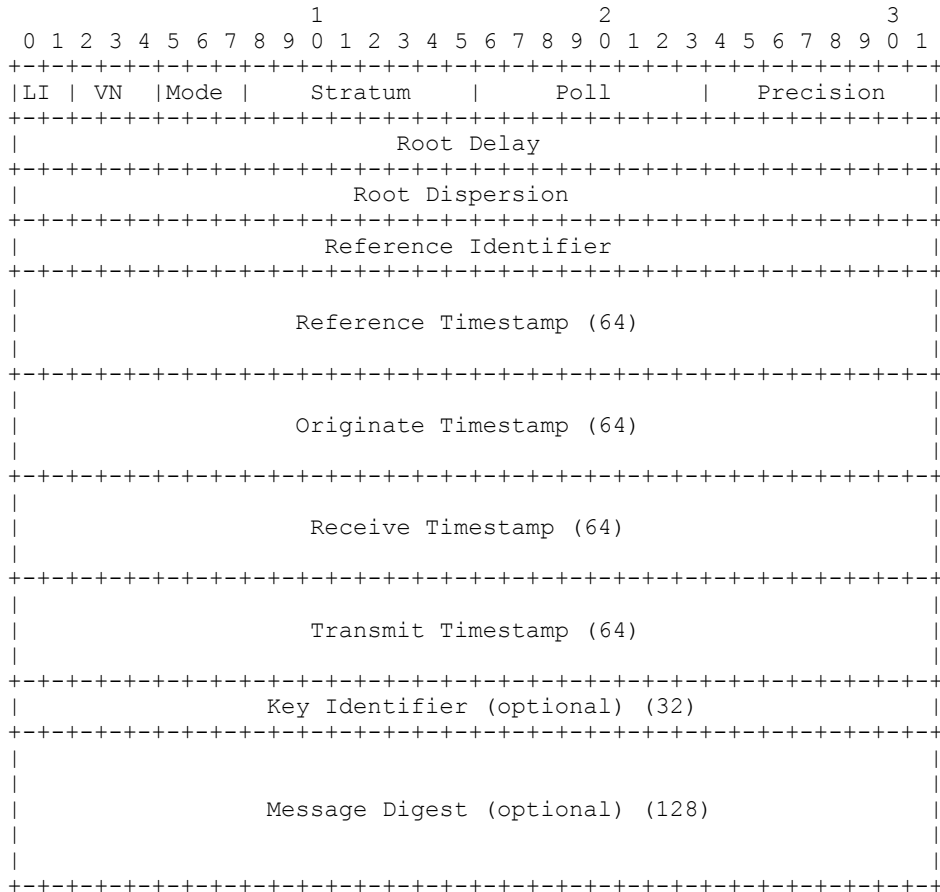
As the NTP timestamp format has been in use for the last 17 years, it remains a possibility that it will be in use 40 years from now when the seconds field overflows. As it is probably inappropriate to archive NTP timestamps before bit 0 was set in 1968, a convenient way to extend the useful life of NTP timestamps is the following convention: If bit 0 is set, the UTC time is in the range 1968-2036 and UTC time is reckoned from 0h 0m 0s UTC on 1 January 1900. If bit 0 is not set, the time is in the range 2036-2104 and UTC time is reckoned from 6h 28m 16s UTC on 7 February 2036. Note that when calculating the correspondence, 2000 is not a leap year. Note also that leap seconds are not counted in the reckoning.

### 12.6.6 NTP Message Format

Both NTP and SNTP are clients of the User Datagram Protocol (UDP)

[POS80], which itself is a client of the Internet Protocol (IP) [DAR81]. The structure of the IP and UDP headers is described in the cited specification documents and will not be detailed further here. The UDP port number assigned to NTP is 123, which should be used in both the Source Port and Destination Port fields in the UDP header. The remaining UDP header fields should be set as described in the specification.

Below is a description of the NTP/SNTP Version 4 message format, which follows the IP and UDP headers. This format is identical to that described in RFC-1305, with the exception of the contents of the reference identifier field. The header fields are defined as follows:



As described in the next section, in SNTP most of these fields are initialized with pre-specified data. For completeness, the function of each field is briefly summarized below.

Leap Indicator (LI): This is a two-bit code warning of an impending leap second to be inserted/deleted in the last minute of the current day, with bit 0 and bit 1, respectively, coded as follows:



LI	Value	Meaning
00	0	no warning
01	1	last minute has 61 seconds
10	2	last minute has 59 seconds)
11	3	alarm condition (clock not synchronized)

**Version Number (VN):** This is a three-bit integer indicating the NTP/SNTP version number. The version number is 3 for Version 3 (IPv4 only) and 4 for Version 4 (IPv4, IPv6 and OSI). If necessary to distinguish between IPv4, IPv6 and OSI, the encapsulating context must be inspected.

**Mode:** This is a three-bit integer indicating the mode, with values defined as follows:

Mode	Meaning
0	reserved
1	symmetric active
2	symmetric passive
3	client
4	server
5	broadcast
6	reserved for NTP control message
7	reserved for private use

In unicast and anycast modes, the client sets this field to 3 (client) in the request and the server sets it to 4 (server) in the reply. In multicast mode, the server sets this field to 5 (broadcast).

**Stratum:** This is a eight-bit unsigned integer indicating the stratum level of the local clock, with values defined as follows:

Stratum	Meaning
0	unspecified or unavailable
1	primary reference (e.g., radio clock)
2-15	secondary reference (via NTP or SNTP)
16-255	reserved

**Poll Interval:** This is an eight-bit signed integer indicating the maximum interval between successive messages, in seconds to the nearest power of two. The values that can appear in this field presently range from 4 (16 s) to 14 (16284 s); however, most applications use only the sub-range 6 (64 s) to 10 (1024 s).

**Precision:** This is an eight-bit signed integer indicating the precision of the local clock, in seconds to the nearest power of two. The values that normally appear in this field range from -6 for mains-frequency clocks to -20 for microsecond clocks found in some workstations.

**Root Delay:** This is a 32-bit signed fixed-point number indicating the total roundtrip delay to the primary reference source, in seconds with fraction point between bits 15 and 16. Note that this variable can take on both positive and negative values, depending on the relative time and frequency offsets. The values that normally appear in this field range from negative values of a few milliseconds to positive values of several hundred milliseconds.

**Root Dispersion:** This is a 32-bit unsigned fixed-point number indicating the nominal error relative to the primary reference source, in seconds with fraction point between bits 15 and 16. The values that normally appear in this field range from 0 to several hundred milliseconds.

Reference Identifier: This is a 32-bit bitstring identifying the particular reference source. In the case of NTP Version 3 or Version 4 stratum-0 (unspecified) or stratum-1 (primary) servers, this is a four-character ASCII string, left justified and zero padded to 32 bits. In NTP Version 3 secondary servers, this is the 32-bit IPv4 address of the reference source. In NTP Version 4 secondary servers, this is the low order 32 bits of the latest transmit timestamp of the reference source. NTP primary (stratum 1) servers should set this field to a code identifying the external reference source according to the following list. If the external reference is one of those listed, the associated code should be used. Codes for sources not listed can be contrived as appropriate.

Code	External Reference Source
LOCL	uncalibrated local clock used as a primary reference for a subnet without external means of synchronization
PPS	atomic clock or other pulse-per-second source individually calibrated to national standards
ACTS	NIST dialup modem service
USNO	USNO modem service
PTB	PTB (Germany) modem service
TDF	Allouis (France) Radio 164 kHz
DCF	Mainflingen (Germany) Radio 77.5 kHz
MSF	Rugby (UK) Radio 60 kHz
WWV	Ft. Collins (US) Radio 2.5, 5, 10, 15, 20 MHz
WWVB	Boulder (US) Radio 60 kHz
WWVH	Kauai Hawaii (US) Radio 2.5, 5, 10, 15 MHz
CHU	Ottawa (Canada) Radio 3330, 7335, 14670 kHz
LORC	LORAN-C radionavigation system
OMEG	OMEGA radionavigation system
GPS	Global Positioning Service
GOES	Geostationary Orbit Environment Satellite

Reference Timestamp: This is the time at which the local clock was last set or corrected, in 64-bit timestamp format.

Originate Timestamp: This is the time at which the request departed the client for the server, in 64-bit timestamp format.

Receive Timestamp: This is the time at which the request arrived at the server, in 64-bit timestamp format.

Transmit Timestamp: This is the time at which the reply departed the server for the client, in 64-bit timestamp format.

Authenticator (optional): When the NTP authentication scheme is implemented, the Key Identifier and Message Digest fields contain the message authentication code (MAC) information defined in Appendix C of RFC-1305.

### 12.6.7 SNTP Client Operations

A SNTP client can operate in multicast mode, unicast mode or anycast mode. In multicast mode, the client sends no request and waits for a broadcast (mode 5) from a designated multicast server. In unicast mode, the client sends a request (mode 3) to a designated unicast server and expects a reply (mode 4) from that server. In anycast mode, the client sends a request (mode 3) to a designated local broadcast or multicast group address and expects a reply (mode 4) from one or more anycast servers. The client uses the first reply received to establish the particular server for subsequent unicast operations. Later replies from this server (duplicates) or any other server are ignored. Other than the selection of address in the request, the operations of anycast and unicast clients are identical. Requests are normally sent at intervals from 64 s to 1024 s, depending on the frequency tolerance of the client clock and the required accuracy.

A unicast or anycast client initializes the NTP message header, sends the request to the server and strips the time of day from the Transmit Timestamp field of the reply. For this purpose, all of the NTP header fields shown above can be set to 0, except the first octet and (optional) Transmit Timestamp fields. In the first octet, the LI field is set to 0 (no warning) and the Mode field is set to 3 (client). The VN field must agree with the version number of the NTP/SNTP server; however, Version 4 servers will also accept previous versions. Version 3 (RFC-1305) and Version 2 (RFC-1119) servers already accept all previous versions, including Version 1 (RFC-1059). Note that Version 0 (RFC-959) is no longer supported by any other version.

Since there will probably continue to be NTP and SNTP servers of all four versions interoperating in the Internet, careful consideration should be given to the version used by SNTP Version 4 clients. It is recommended that clients use the latest version known to be supported by the selected server in the interest of the highest accuracy and reliability. SNTP Version 4 clients can interoperate with all previous version NTP and SNTP servers, since the header fields used by SNTP clients are unchanged. Version 4 servers are required to reply in the same version as the request, so the VN field of the request also specifies the version of the reply.

While not necessary in a conforming client implementation, in unicast and anycast modes it is highly recommended that the transmit timestamp in the request is set to the time of day according to the client clock in NTP timestamp format. This allows a simple calculation to determine the propagation delay between the server and client and to align the local clock generally within a few tens of milliseconds relative to the server. In addition, this provides a simple method to verify that the server reply is in fact a legitimate response to the specific client request and avoid replays. In multicast mode, the client has no information to calculate the propagation delay or determine the validity of the server, unless the NTP authentication scheme is used.

To calculate the roundtrip delay  $d$  and local clock offset  $t$  relative to the server, the client sets the transmit timestamp in the request to the time of day according to the client clock in NTP timestamp format. The server copies this field to the originate timestamp in the reply and sets the receive timestamp and transmit timestamp to the time of day according to the server clock in NTP timestamp format.

When the server reply is received, the client determines a Destination Timestamp variable as the time of arrival according to its clock in NTP timestamp format. The following table summarizes the four timestamps.

Timestamp Name	ID	When Generated
Originate Timestamp	T1	time request sent by client
Receive Timestamp	T2	time request received by server
Transmit Timestamp	T3	time reply sent by server
Destination Timestamp	T4	time reply received by client

The roundtrip delay  $d$  and local clock offset  $t$  are defined as

$$d = (T4 - T1) - (T2 - T3) \quad t = ((T2 - T1) + (T3 - T4)) / 2.$$

The following table summarizes the SNTP client operations in unicast, anycast and multicast modes. The recommended error checks are shown in the Reply and Multicast columns in the table. The message should be considered valid only if all the fields shown contain values in the respective ranges. Whether to believe the message if one or more of the fields marked “ignore” contain invalid values is at the discretion of the implementation.

Field Name	Unicast/Anycast	Multicast	
	Request	Reply	
LI	0	0-2	0-2
VN	1-4	copied from request	1-4
Mode	3	4	5
Stratum	0	1-14	1-14
Poll	0	ignore	ignore
Precision	0	ignore	ignore
Root Delay	0	ignore	ignore
Root Dispersion	0	ignore	ignore
Reference Identifier	0	ignore	ignore
Reference Timestamp	0	ignore	ignore
Originate Timestamp	0	(see text)	ignore
Receive Timestamp	0	(see text)	ignore
Transmit Timestamp	(see text)	nonzero	nonzero
Authenticator	optional	optional	optional

### 12.6.8 SNTP Server Operations

A SNTP Version 4 server operating with either a NTP or SNTP client of the same or previous versions retains no persistent state. Since a SNTP server ordinarily does not implement the full set of NTP algorithms intended to support redundant peers and diverse network paths, a SNTP server should be operated only in conjunction with a source of external synchronization, such as a reliable radio clock or telephone modem. In this case it always operates as a primary (stratum 1) server.

A SNTP server can operate in unicast mode, anycast mode, multicast mode or any combination of these modes. In unicast and anycast modes, the server receives a request (mode 3), modifies certain fields in the NTP header, and sends a reply (mode 4), possibly using the same message buffer as the request. In anycast mode, the server listens on the designated local broadcast or multicast group address assigned by the IANA, but uses its own unicast address in the source address field of the reply. Other than the selection of address in the reply, the operations of anycast and unicast servers are identical. Multicast messages are normally sent at poll intervals from 64 s to 1024 s, depending on the expected frequency tolerance of the client clocks and the required accuracy.

In unicast and anycast modes, the VN and Poll fields of the request are copied intact to the reply. If the Mode field of the request is 3 (client), it is set to 4 (server) in the reply; otherwise, this field is set to 2 (symmetric passive) in order to conform to the NTP specification. This allows clients configured in symmetric active (mode 1) to interoperate successfully, even if configured in possibly suboptimal ways. In multicast (unsolicited) mode, the VN field is set to 4, the Mode field is set to 5 (broadcast), and the Poll field set to the nearest integer base-2 logarithm of the poll interval.

Note that it is highly desirable that, if a server supports multicast mode, it also supports unicast mode. This is so a potential multicast client can calculate the propagation delay using a client/server exchange prior to regular operation using only multicast mode. If the server supports anycast mode, then it must support unicast mode. There does not seem to be a great advantage to operate both multicast and anycast modes at the same time, although the protocol specification does not forbid it.

In unicast and anycast modes, the server may or may not respond if not synchronized to a correctly operating radio clock, but the preferred option is to respond, since this allows reachability to be determined regardless of synchronization state. In multicast mode, the server sends broadcasts only if synchronized to a correctly operating reference clock.

The remaining fields of the NTP header are set in the following way. Assuming the server is synchronized to a radio clock or other primary reference source and operating correctly, the LI field is set to 0 and the Stratum field is set to 1 (primary server); if not, the Stratum field is set to 0 and the LI field is set to 3. The Precision field is set to reflect the maximum reading error of the local clock. For all practical cases it is computed as the negative of the number of significant bits to the right of the decimal point in the NTP timestamp format. The Root Delay and Root Dispersion fields are set to 0 for a primary server; optionally, the Root Dispersion field can be set to a value corresponding to the maximum expected error of the radio clock itself. The Reference Identifier is set to designate the primary reference source, as indicated in the table of Section 5 of this document.

The timestamp fields are set as follows. If the server is unsynchronized or first coming up, all timestamp fields are set to zero. If synchronized, the Reference Timestamp is set to the time the last update was received from the radio clock or modem. In unicast and anycast modes, the Receive Timestamp and Transmit Timestamp fields are set to the time of day when the message is sent and the Originate Timestamp field is copied unchanged from the Transmit Timestamp field of the request. It is important that this field be copied intact, as a NTP client uses it to avoid replays. In multicast mode, the Originate Timestamp and Receive Timestamp fields are set to 0 and the Transmit Timestamp field is set to the time of day when the message is sent. The following table summarizes these actions.

Field Name	Unicast/Anycast		Multicast
	Request	Reply	
LI	ignore	0 or 3	0 or 3
VN	1-4	copied from request	4
Mode	3	2 or 4	5
Stratum	ignore	1	1
Poll	ignore	copied from request	log2 poll interval
Precision	ignore	-log2 server significant bits	-log2 server significant bits
Root Delay	ignore	0	0
Root Dispersion	ignore	0	0
Reference Identifier	ignore	source ident	source ident
Reference Timestamp	ignore	time of last radio update	time of last radio update
Originate Timestamp	ignore	copied from transmit timestamp	0

Receive Timestamp	ignore	time of day	0
Transmit Timestamp	(see text)	time of day	time of day
Authenticator	optional	optional	optional

There is some latitude on the part of most clients to forgive invalid timestamps, such as might occur when first coming up or during periods when the primary reference source is inoperative. The most important indicator of an unhealthy server is the LI field, in which a value of 3 indicates an unsynchronized condition. When this value is displayed, clients should discard the server message, regardless of the contents of other fields.

### 12.6.9 Configuration and Management

Initial setup for SNTP servers and clients can be done using a configuration file if a file system is available, or a serial port if not. It is intended that in-service management of NTP and SNTP Version 4 servers and clients be performed using SNMP and a suitable MIB to be published later. Ordinarily, SNTP servers and clients are expected to operate with little or no site-specific configuration, other than specifying the IP address and subnet mask or OSI NSAP address.

Unicast clients must be provided with the designated server name or address. If a server name is used, the address of one of more DNS servers must be provided. Multicast servers and anycast clients must be provided with the TTL and local broadcast or multicast group address. Anycast servers and multicast clients may be configured with a list of address-mask pairs for access control, so that only those clients or servers known to be trusted will be used. These servers and clients must implement the IGMP protocol and be provided with the local broadcast or multicast group address as well. The configuration data for cryptographic authentication is beyond the scope of this document.

There are several scenarios which provide automatic server discovery and selection for SNTP clients with no pre-specified configuration, other than the IP address and subnet mask or OSI NSAP address. For a IP subnet or LAN segment including a fully functional NTP server, the clients can be configured for multicast mode using the local broadcast address. The same approach can be used with other servers using the multicast group address. In both cases, provision of an access control list is a good way to insure only trusted sources can be used to set the local clock.

In another scenario suitable for an extended network with significant network propagation delays, clients can be configured for anycast mode, both upon initial startup and after some period when the currently selected unicast source has not been heard. Following the defined protocol, the client binds to the first reply heard and continues operation in unicast mode. In this mode the local clock can be automatically adjusted to compensate for the propagation delay.

In still another scenario suitable for any network and where multicast service is not available, the DNS can be set up with a common CNAME, like `time.domain.net`, and a list of address records for NTP servers in the same domain. Upon resolving `time.domain.net` and obtaining the list, the client selects a server at random and begins operation in unicast mode with that server. Many variations on this theme are possible.

### 12.6.10 Acknowledgements

Jeff Learman was helpful in developing the OSI model for this protocol. Ajit Thyagarajan provided valuable suggestions and corrections.

### 12.6.11 References

- [COL94] Colella, R., R. Callon, E. Gardner, Y. Rekhter, "Guidelines for OSI NSAP allocation in the Internet", RFC 1629, NIST, May 1994.
- [DAR81] Postel, J., "Internet Protocol", STD 5, RFC 791, USC Information Sciences Institute, September 1981.

- [DEE89] Deering, S., "Host extensions for IP multicasting", STD 5, RFC 1112, Stanford University, August 1989.
- [DEE96] Deering, S., R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 1883, Xerox and Ipsilon, January 1996.
- [DOB91] Dobbins, K, W. Haggerty, C. Shue, "OSI connectionless transport services on top of UDP - Version: 1", RFC 1240, Open Software Foundation, June 1991.
- [EAS95] Eastlake, D., 3rd., and C. Kaufman, "Domain Name System Security Extensions", Work in Progress.
- [FUR94] Furniss, P., "Octet sequences for upper-layer OSI to support basic communications applications", RFC 1698, Consultant, October 1994.
- [HIN96] Hinden, R., and S. Deering, "IP Version 6 addressing Architecture", RFC 1884, Ipsilon and Xerox, January 1996.
- [ISO86] International Standards 8602 - Information Processing Systems - OSI: Connectionless Transport Protocol Specification. International Standards Organization, December 1986.
- [MIL92] Mills, D., "Network Time Protocol (Version 3) specification, implementation and analysis", RFC 1305, University of Delaware, March 1992.
- [PAR93] Partridge, C., T. Mendez and W. Milliken, "Host anycasting service", RFC 1546, Bolt Beranek Newman, November 1993.
- [POS80] Postel, J., "User Datagram Protocol", STD 6, RFC 768, USC Information Sciences Institute, August 1980.
- [POS83] Postel, J., "Time Protocol", STD 26, RFC 868, USC Information Sciences Institute, May 1983.

#### Security Considerations

Security issues are not discussed in this memo.

#### Author's Address

David L. Mills  
Electrical Engineering Department  
University of Delaware  
Newark, DE 19716

Phone: (302) 831-8247

## **12.7 RFC867 (DAYTIME)**

Network Working Group J. Postel - ISI

Request for Comments: 867

May 1983

### **Daytime Protocol**

This RFC specifies a standard for the ARPA Internet community. Hosts on the ARPA Internet that choose to implement a Daytime Protocol are expected to adopt and implement this standard.

A useful debugging and measurement tool is a daytime service. A daytime service simply sends a the current date and time as a character string without regard to the input.

#### **12.7.1 TCP Based Daytime Service**

One daytime service is defined as a connection based application on TCP. A server listens for TCP connections on TCP port 13. Once a connection is established the current date and time is sent out the connection as a ASCII character string (and any data received is thrown away). The service closes the connection after sending the quote.

#### **12.7.2 UDP Based Daytime Service**

Another daytime service is defined as a datagram based application on UDP. A server listens for UDP datagrams on UDP port 13. When a datagram is received, an answering datagram is sent containing the current date and time as a ASCII character string (the data in the received datagram is ignored).

#### **12.7.3 Daytime Syntax**

There is no specific syntax for the daytime. It is recommended that it be limited to the ASCII printing characters, space, carriage return, and line feed. The daytime should be just one line.

One popular syntax is:

Weekday, Month Day, Year Time-Zone

Example:

Tuesday, February 22, 1982 17:37:43-PST

Another popular syntax is that used in SMTP:

dd mmm yy hh:mm:ss zzz

Example:

02 FEB 82 07:59:01 PST

NOTE: For machine useful time use the Time Protocol (RFC868).



## 12.8 RFC868 (TIME)

Network Working Group J. Postel - ISI

Request for Comments: 868 K. Harrenstien - SRI

May 1983

### Time Protocol

This RFC specifies a standard for the ARPA Internet community. Hosts on the ARPA Internet that choose to implement a Time Protocol are expected to adopt and implement this standard.

This protocol provides a site-independent, machine-readable date and time. The Time service sends back to the originating source the time in seconds since midnight on January first 1900.

One motivation arises from the fact that not all systems have a date/time clock, and all are subject to occasional human or machine error. The use of time-servers makes it possible to quickly confirm or correct a system's idea of the time, by making a brief poll of several independent sites on the network.

This protocol may be used either above the Transmission Control Protocol (TCP) or above the User Datagram Protocol (UDP).

When used via TCP the time service works as follows:

```
S Listen on port 37 (45 octal).
U Connect to port 37.
S Send the time as a 32 bit binary number.
U Receive the time.
U Close the connection.
S Close the connection.
```

The server listens for a connection on port 37. When the connection is established, the server returns a 32-bit time value and closes the connection. If the server is unable to determine the time at its site, it should either refuse the connection or close it without sending anything.

When used via UDP the time service works as follows:

```
S Listen on port 37 (45 octal).
U Send an empty datagram to port 37.
S Receive the empty datagram.
S Send a datagram containing the time as a 32 bit binary number.
U Receive the time datagram.
```

The server listens for a datagram on port 37. When a datagram arrives, the server returns a datagram containing the 32-bit time value. If the server is unable to determine the time at its site, it should discard the arriving datagram and make no reply.

#### 12.8.1 The Time

The time is the number of seconds since 00:00 (midnight) 1 January 1900 GMT, such that the time 1 is 12:00:01 am on 1 January 1900 GMT; this base will serve until the year 2036.

For example:

the time 2,208,988,800 corresponds to 00:00 1 Jan 1970 GMT,

2,398,291,200 corresponds to 00:00 1 Jan 1976 GMT,

2,524,521,600 corresponds to 00:00 1 Jan 1980 GMT,

2,629,584,000 corresponds to 00:00 1 May 1983 GMT,

and -1,297,728,000 corresponds to 00:00 17 Nov 1858 GMT.

## 12.9 RFC792 (ICMP Timestamp section)

Network Working Group  
Request for Comments: 792

J. Postel  
ISI  
ember 1981

Updates: RFCs 777, 760  
Updates: IENs 109, 128

# INTERNET CONTROL MESSAGE PROTOCOL

## DARPA INTERNET PROGRAM PROTOCOL SPECIFICATION

### Introduction

The Internet Protocol (IP) [1] is used for host-to-host datagram service in a system of interconnected networks called the Catenet [2]. The network connecting devices are called Gateways. These gateways communicate between themselves for control purposes via a Gateway to Gateway Protocol (GGP) [3,4]. Occasionally a gateway or destination host will communicate with a source host, for example, to report an error in datagram processing. For such purposes this protocol, the Internet Control Message Protocol (ICMP), is used. ICMP, uses the basic support of IP as if it were a higher level protocol, however, ICMP is actually an integral part of IP, and must be implemented by every IP module.

ICMP messages are sent in several situations: for example, when a datagram cannot reach its destination, when the gateway does not have the buffering capacity to forward a datagram, and when the gateway can direct the host to send traffic on a shorter route.

The Internet Protocol is not designed to be absolutely reliable. The purpose of these control messages is to provide feedback about problems in the communication environment, not to make IP reliable. There are still no guarantees that a datagram will be delivered or a control message will be returned. Some datagrams may still be undelivered without any report of their loss. The higher level protocols that use IP must implement their own reliability procedures if reliable communication is required.

The ICMP messages typically report errors in the processing of datagrams. To avoid the infinite regress of messages about messages etc., no ICMP messages are sent about ICMP messages. Also ICMP messages are only sent about errors in handling fragment zero of fragmented datagrams. (Fragment zero has the fragment offset equal zero).

### Message Formats

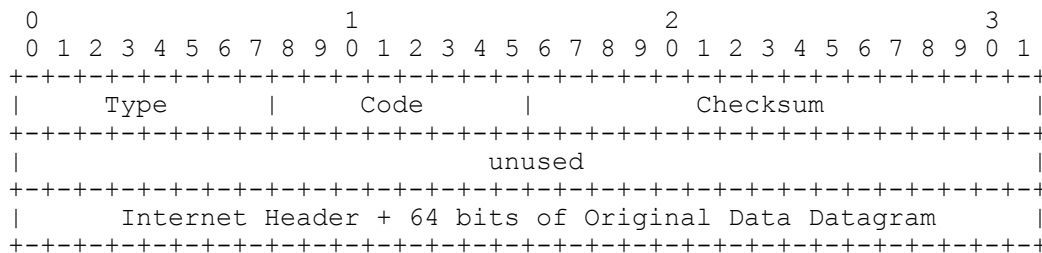
ICMP messages are sent using the basic IP header. The first octet of the data portion of the datagram is a ICMP type field; the value of this field determines the format of the remaining data. Any field labeled "unused" is reserved for later extensions and must be zero when sent, but receivers should not use these fields (except to include them in the checksum). Unless otherwise noted under the individual format descriptions, the values of the internet header fields are as follows:

Version	4
IHL	Internet header length in 32-bit words.
Type of Service	0
Total Length	Length of internet header and data in octets.

Identification, Flags, Fragment Offset	Used in fragmentation, see [1].
Time to Live	Time to live in seconds; as this field is decremented at each machine in which the datagram is processed, the value in this field should be at least as great as the number of gateways which this datagram will traverse
Protocol	ICMP = 1
Header Checksum	The 16 bit one's complement of the one's complement sum of all 16 bit words in the header. For computing the checksum, the checksum field should be zero. This checksum may be replaced in the future.
Source Address	The address of the gateway or host that composes the ICMP message. Unless otherwise noted, this can be any of a gateway's addresses.
Destination Address	The address of the gateway or host to which the message should be sent.

[section of RFC removed]...

## Timestamp or Timestamp Reply Message



## IP Fields:

Addresses

The address of the source in a timestamp message will be the destination of the timestamp reply message. To form a timestamp reply message, the source and destination addresses are simply reversed, the type code changed to 14, and the checksum recomputed.

## ICMP Fields:

Type	13 for timestamp message; 14 for timestamp reply message.
Code	0
Checksum	The checksum is the 16-bit ones's complement of the one's complement sum of the ICMP message starting with the ICMP Type. For computing the checksum, the checksum field should be zero. This checksum may be replaced in the future.
Identifier	If code = 0, an identifier to aid in matching timestamp and replies, may be zero.
Sequence Number	If code = 0, a sequence number to aid in matching timestamp and replies, may be zero.

## Description

The data received (a timestamp) in the message is returned in the reply together with an additional timestamp. The timestamp is 32 bits of milliseconds since midnight UT. One use of these timestamps is described by Mills [5].

The Originate Timestamp is the time the sender last touched the message before sending it, the Receive Timestamp is the time the echoer first touched it on receipt, and the Transmit Timestamp is the time the echoer last touched the message on sending it.

If the time is not available in milliseconds or cannot be provided with respect to midnight UT then any time can be inserted in a timestamp provided the high order bit of the timestamp is also set to indicate this non-standard value.

The identifier and sequence number may be used by the echo sender to aid in matching the replies with the requests. For example, the identifier might be used like a port in TCP or UDP to identify a session, and the sequence number might be incremented on each request sent. The destination returns these same values in the reply.

Code 0 may be received from a gateway or a host.

[section of RFC removed]...

## Summary of Message Types

0	Echo Reply
3	Destination Unreachable
4	Source Quench
5	Redirect
8	Echo
11	Time Exceeded
12	Parameter Problem
13	Timestamp
14	Timestamp Reply
15	Information Request
16	Information Reply

## References

- [1] Postel, J. (ed.), "Internet Protocol - DARPA Internet Program Protocol Specification," [RFC 791](#), USC/Information Sciences Institute, September 1981.
- [2] Cerf, V., "The Catenet Model for Internetworking," IEN 48, Information Processing Techniques Office, Defense Advanced Research Projects Agency, July 1978.
- [3] Strazisar, V., "Gateway Routing: An Implementation Specification", IEN 30, Bolt Beranek and Newman, April 1979.
- [4] Strazisar, V., "How to Build a Gateway", IEN 109, Bolt Beranek and Newman, August 1979.
- [5] Mills, D., "DCNET Internet Clock Service," [RFC 778](#), COMSAT Laboratories, April 1981.

### **12.10 Other radio and GPS clocks supported.**

Tardis supports the following radio and GPS clocks. They can all be connected to a serial port on the PC.

- Trimble Palisade
- EndRun Technologies Præcis Ct
- Kinometrics Truetime
- Spectracom WWVB
- Neol NeoCLock
- ExpertMouse Clock
- EMC Professional in XNTP mode
- Radio pulse clock connected to DCD pin of serial port
- IRIG-B