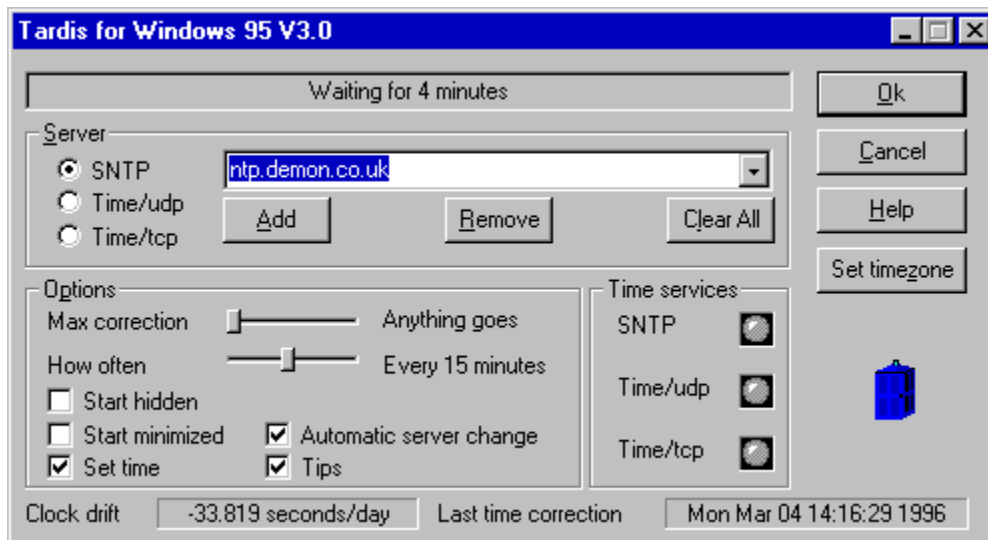## Tardis for Windows 95 version 3.0

Tardis is a utility for Windows 95 that corrects your PC's clock.   It does it by using a networked time server.   Tardis requires TCP/IP to be installed.   A typical use of Tardis is to synchronise the PC's clock on starting a SLIP/PPP connection to an Internet provider.   Another is to have it running all the time in a LAN environment correcting the clock once an hour.   Tardis version 3.0 also provides time servers to allow PC's to synchronise their clocks even where no other time server is present.

Tardis can use the time protocols specified in <u>RFC868</u> (Time) and <u>RFC1361</u>(SNTP) as both a client and a server.

<u>Note: Tardis version 3.0 is shareware</u>

Click on the dialog box for help on Tardis settings.

If this is *not* set Tardis will *not* set the system time, useful 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.

If this is *set* Tardis will hide itself after starting up.   You can show it and hide it again by clicking on the Tardis Icon in the system tray.

Select the protocol used by your time server here.   SNTP is best if your server supports it.   Otherwise use either of the Time protocols.   The Time/udp uses very little network bandwidth.

If this is set then Tardis will start in a minimised state.

This shows the last time that Tardis set the time on this PC.

This is an estimate of how much your PC's clock drifts.   Treat this estimate with a pinch of salt.   It will 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.   The Time protocol isn't accurate enough.

This setting controls whether tooltips are shown.   These are useful one line prompts that pop up when the mouse is placed over setting.

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 pull down list.

Select your timezone here.   This button takes you to the Control Panel for Date and Time

Tardis 95 version 3.0 provides servers for all the time protocols that it supports. I.e. you can run Tardis 95 on a PC and other PC's will be able to retrieve the time from it.   This enables you to lock PC's clocks together so that they always agree even in a local LAN with no other time sources.
When a time service is accessed from another machine the appropriate LED will flash to indicate activity.

This setting is the name or 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.

There are many good timeservers on the Internet, try to choose one near to you.

The server machine will often be a Unix™ machine, but a version of Tardis is available for Windows NT 3.1/3.5 that does the same job as this version *but* also acts a time server.   Available at all good NT archive sites.   The server may also be another PC running Tardis 95 version 3.0.

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.   You can also tell Tardis to exit once it has read the time.   If it can't it get the time it will display the status and try again.   This is useful for setting the time when first establishing a dial-up Internet connection.

This shows what Tardis is up to.   The text will change to provide useful progress and error messages.

When you press OK the settings currently shown become active.

When you press the help button this help should be displayed.

When you press Cancel the settings currently shown are replaced with those last saved by pressing OK. Useful if you fiddled with the settings and then decided not to do it after all.

This is meant to be a cute icon showing a little picture of Dr Who's Tardis after which this program is named.   It isn't as good as the real thing. Dr. Who is a very long running SciFi programme on BBC TV.

These buttons allow you to add and remove sites from the list.   The *add* button adds the current site shown and *remove* removes it.   *Clear all* is a quick way to remove all the sites.

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 setting specifies what is reasonable.   This is useful if your timezone is wrong or your time server has gone mad.

It is also possible to specify   that any correction is allowed.

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.

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.

Shareware

Copyright Notice

Tardis for Windows 95 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 US$20 (or the rough equivalent in your local currency) to be sent by post to:

H. C. Mingham-Smith
33 Arthur Rd.
Wokingham,
Berkshire   RG11 2SS
England.

A cheque made payable to H.C. Mingham-Smith would be acceptable.
For sites where multiple copies are used I would request US$40 for the right to use on any machine at the site.   Source licenses are also available.

Please send e-mail regarding Tardis to tardis@kaska.demon.co.uk.

Simple Network Time Protocol (SNTP)

Status of this Memo

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

Abstract

This memorandum describes the Simple Network Time Protocol (SNTP), 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. It involves no change to the current or previous NTP specification versions or known implementations, but rather a clarification of certain design features of NTP which allow operation in a simple, stateless RPC mode with accuracy and reliability expectations similar to the UDP/TIME protocol described in RFC-868.

This memorandum does not obsolete or update any RFC. A working knowledge of RFC-1305 is not required for an implementation of SNTP.

1.   Introduction

The Network Time Protocol (NTP) specified in RFC-1305 [MIL92] is 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 jitter characteristics of the synchronization source and network paths.

RFC-1305 specifies the NTP protocol machine in terms of events, states, transition functions and actions and, in addition, optional algorithms to improve the timekeeping quality and mitigate among several, possibly faulty, synchronization sources. 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 of this order are not required and something less, perhaps in the order of one second, is sufficient. In such cases simpler protocols such as the Time Protocol [POS83], have been used for this purpose. These protocols usually involve a remote-procedure call (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 members of the Internet NTP synchronization subnet of today use software packages including the full suite of NTP options and algorithms, which are relatively complex, real-time applications. While the software has been ported to a wide variety of hardware platforms ranging from supercomputers to personal computers, its sheer size and complexity is not appropriate for many applications. Accordingly, it is useful to explore alternative access strategies using far simpler software appropriate for accuracy expectations in the order of a second.

This memorandum describes the Simple Network Time Protocol (SNTP), which is a simplified access strategy for servers and clients using NTP as now specified and deployed in the Internet. There are no changes to the protocol or implementations now running or likely to be implemented in the near future. 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.

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 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 clock 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 clocks and backup paths to other primary servers should the radio clock fail or become faulty. Therefore, the use of SNTP rather than NTP in primary servers should be carefully considered.

2.   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 zero 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 zero preceding bit zero.

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. This format allows convenient multiple-precision arithmetic and conversion to Time Protocol representation (seconds), but does complicate the conversion to ICMP Timestamp message representation (milliseconds). The precision of this representation is about 200 picoseconds, which should be adequate for even the most exotic requirements.

Note that since some time in 1968 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. 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).   Timestamped data requiring such qualification will be so precious that appropriate means should be readily available. There will exist a 200-picosecond interval, henceforth ignored, every 136 years when the 64-bit field will be zero, which by convention is interpreted as an invalid timestamp.

3.   NTP Message Format

Both NTP and SNTP are clients of the User Datagram Protocol (UDP) [POS83], which itself is a client of the Internet Protocol (IP) [DAR81]. The structure of the IP and UDP headers is described in the relevant specification documents and will not be described further in this memorandum. Following is a description of the SNTP message format, which follows the IP and UDP headers. The SNTP message format is identical to the NTP format described in RFC-1305, with the exception that some of the data fields are "canned," that is, initialized to prespecified values. The format of the NTP message data area, which immediately follows the UDP header, is shown below.

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
6 7 8 9 0 1
|LI | VN  |Mode |    Stratum    |     Poll      |
Precision   |
                    Root Delay
                 Root Dispersion
              Reference Identifier
            Reference Timestamp (64)
            Originate Timestamp (64)
             Receive Timestamp (64)
            Transmit Timestamp (64)
         Authenticator (optional) (96)
```

As described in the next section, in SNTP most of these fields are initialized with prespecified 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 version number, currently 3.

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

The use of this field will be discussed in the next section.

Stratum: This is a eight-bit 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 normally appear in this field range from 6 to 10, inclusive.

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 -18 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 errors. 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 maximum 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 zero to several hundred milliseconds.

Reference Clock Identifier: This is a 32-bit code identifying the particular reference clock. In the case of stratum 0 (unspecified) or stratum 1 (primary reference), this is a four-octet, left-justified, zero-padded ASCII string. While not enumerated as part of the NTP specification, the following are representative ASCII identifiers:

Stratum	Code	Meaning
0	ascii	generic time service other than NTP, such as ACTS (Automated Computer Time Service), TIME (UDP/Time Protocol), TSP (TSP Unix time protocol), DTSS(Digital Time Synchronization Service), etc.
1	ATOM	calibrated atomic clock
1	VLF	VLF radio (OMEGA, etc.)
1	callsign	Generic radio
1	LORC	LORAN-C radionavigation system
1	GOES	Geostationary Operational Environmental Satellite
1	GPS	Global Positioning Service
2	address	secondary reference (four-octet Internet address of the NTP server)

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

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

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

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

Authenticator (optional): When the NTP authentication mechanism is implemented, this contains the authenticator information defined in Appendix C of RFC-1305. In SNTP this field is ignored for incoming messages and is not generated for outgoing messages.

4.   SNTP Client Operations

The model for an SNTP client operating with either an NTP or SNTP server is a RPC client with no persistent state. The client initializes the SNTP message header, sends the message to the server and strips the time of day from the reply. For this purpose all of the message-header fields shown above are set to zero, except the first octet. In this octet the Leap Indicator is set to zero (no warning) and the Mode to 3 (client). The Version Number must agree with the software version of the NTP or SNTP server; however, NTP Version 3 (RFC-1305) servers will also accept Version 2 (RFC-1119) and Version 1 (RFC-1059) messages, while NTP Version 2 servers will also accept NTP Version 1 messages. Version 0 (original NTP described in RFC-959) messages are no longer supported. Since there are NTP servers of all three versions operating in the Internet of today, it is recommended that the Version Number field be set to one.

The server reply includes all the fields described above; however, in SNTP only the Transmit Timestamp has explicit meaning. The integer part of this field contains the server time of day in the same format as the Time Protocol. While the fraction part of this field will usually be valid, the accuracy achieved with the SNTP mode of access probably does not justify

its use.

The following table is a summary of the SNTP client operations. There are three recommended error checks shown in the table. In all NTP versions, if the Leap Indicator field is 3 or the Transmit Timestamp is zero (unsynchronized), the server has never synchronized or not synchronized to a valid timing source within the last 24 hours. If the Stratum field is 0 (unspecified or unavailable), the server has never synchronized, has lost reachability with all timing sources or is synchronized by some protocol other than NTP. Whether to believe the transmit timestamp or not in this case is at the discretion of the client implementation.

Field Name	Request	Reply
Leap Indicator (LI)	0	if 3 (unsynchronized),disregard
Version Number (VN)	(see text)	ignore
Mode	3	(client) ignore
Stratum	0	if 0 (unspecified), disregard
Poll	0	ignore
Precision	0	ignore
Root Delay	0	ignore
Root Dispersion	0	ignore
Reference Identifier	0	ignore
Reference Timestamp	0	ignore
Originate Timestamp	0	ignore
Receive Timestamp	0	ignore
Transmit Timestamp	0	time of day (seconds only); if 0 (unsynchronized), disregard
Authenticator	(not used)	ignore

5.   SNTP Server Operations

The model for an SNTP server operating with either an NTP or SNTP client is an RPC server with no persistent state. The SNTP server ignores all header fields except the first octet, modifies certain fields and returns the message to the sender. Since an SNTP server ordinarily does not implement the full set of NTP algorithms intended to support the highest quality service, it is recommended that an SNTP server be operated only in conjunction with a source of outside synchronization, such as a radio clock. In this case the server always operates at stratum 1.

The first octet is interpreted as follows. The Leap Indicator and Version Number fields are ignored. Optionally, messages with version numbers other than 1, 2, or 3 can be discarded. For primary servers connected to a functioning radio clock, the Leap Indicator field is set to zero and the Stratum field is set to one in the reply.   otherwise, these fields are set to 3 and zero, respectively. In any case the Version Number and Poll fields are copied intact to the reply message header. If The Mode field is set to 3 (client), it is changed to 4 (server) in the reply; otherwise, this field is set to 2 (symmetric passive).

The Stratum 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 zero for a primary server; optionally, the Root Dispersion can be set to a value corresponding to the expected (constant) maximum expected error of the primary reference source.   The Reference Identifier is set to designate the primary reference source, as indicated in the table above. If this information is unspecified or unavailable, the field is set to zero.

The timestamp fields are set as follows. The Reference Timestamp, Receive Timestamp and Transmit Timestamp fields are set to the time of day at the server. The Originate Timestamp field is copied unchanged from the request. The following table summarizes these actions.

Field Name	Request	Reply
Leap Indicator (LI)	ignore	0 (normal), 3(unsynchronized)
Version Number (VN)	ignore	copied from request
Mode	(see text)	(see text)
Stratum	ignore	server stratum (1)
Poll	ignore	copied from request
Precision	ignore	server precision
Root Delay	ignore	0
Root Dispersion	ignore	0 (see text)
Reference Identifier	ignore	source identifier or 0
Reference Timestamp	ignore	time of day or 0
Originate Timestamp	ignore	copied from request
Receive Timestamp	ignore	time of day or 0
Transmit Timestamp	ignore	time of day or 0

Authenticator   ignore   (not used)

6.   References

[DAR81] Postel, J., "Internet Protocol - DARPA Internet Program Protocol Specification", RFC 791, DARPA, September 1981.

[MIL92] Mills, D., "Network Time Protocol (Version 3) Specification, Implementation and Analysis", RFC 1305, University of Delaware, March 1992.

[POS80] Postel, J., "User Datagram Protocol", RFC 768, USC/Information Sciences Institute, August 1980.

[POS83] Postel, J., and K. Harrenstien, "Time Protocol", RFC 868, USC/Information Sciences Institute, SRI, 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
EMail: mills@udel.edu