

Tardis time service

Tardis is a Windows NT service that synchronises your PCs clock to a central time Server.

The Tardis control panel allows the Server to be specified, the number of minutes between successive synchronisations, and the amount of correction allowed.

Dont forget to set the correct timezone in the Date/Time control panel!

Registration

Server

The server machine will typically be a Unix™ machine running a time server. Most do as standard. Tardis requires the Windows NT TCP/IP protocol software to be installed on the PC. The time is obtained using the RFC868 protocol.

The server address may be entered as a name, e.g. hinge.demon.co.uk, or as an Internet address e.g. 123.123.123.123.

Tardis will also *act* as a server for the time. i.e. Other machines can ask Tardis for the time. This also conforms to the RFC868 protocol.

Synchronise every n minutes

Tardis will contact the server and get a new time every so often. This setting tells tardis how often it should be done. 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, but dont forget that your time server may be down for those few crucial minutes.

Amount of correction allowed

Tardis validates the time it got 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. A setting of 3600 would allow a correction of ± 1 hour. Any more than this is likely to be a problem with timezones rather than a drifting clock.

Registration

Tardis is shareware. Please see the manual for details of how to register.

RFC868

Network Working Group
Request for Comments: 868

J. Postel - ISI
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.

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.

