SNTP Client for Netware V1.6

What is SNTPCLNT

SNTPCLNT is a SNTP client for Netware 3.x,4.x and 5.x Servers. SNTP stands for Simple Network Time Protocol, and should not be confused with the SMTP or SNMP protocols. Have a look at RFC 2030 for a detailed discussion of the protocol.

What can SNTPCLNT do for you ?

Imagine you have a direct connection to the internet, or you have a Unix system with a atomic clock. Now, the hardwareclock of your novellserver is not very exact, and you have to readjust the time all 2-3 weeks.

So would it not be nice to ask a host on the inter/intra-net about the current time/date and adjust the server according to this answer ???

If the answer is yes, then you need my utility.

What do you need to use this utility ?

- Netware 3.x, 4.x or 5.x server with TCP/IP installed
- Direct connection to the internet (or to a (S)NTP server)
- My utility, either the Netware 3.x,4.x or 5.x version
- Optionally sntpclnt can benefit of an installed DNS resolver (only 4.x/5.x version)

Quickstart to use SNTPCLNT

First you must decide whether you have a 3.x, 4.x or 5.x server. If you don't know which version of server you have ;-), use the version for the 3.x servers, since this works on 4.x/5.x servers too.
Then you copy the SNTPCLNT.CFG file in the SYS:ETC directory and customize it to your needs.
You should at least modify the NTP servers. For a list of available NTP servers have a look at ???.
You will then have to obtain a valid ActivationKey for the NLM. You can get a 30 day trialkey at www.neatech.ch/sntpclnt.

SNTPCLNT will now contact the given host(s), and set your clock accordingly. If you have specified an interval, then the NTP server(s) will be polled again, and again and again and again and.....

Since your server is now "up-to-date" you have time to read the rest of this document.

What's new in the 1.6 release ?

There are only minor corrections made to the NLM. One of them solving a possible abend when no statistic screen had been defined.
It is now simpler to operate it with older NTP servers, since it autodetects the version of the servers.

Detailed description of SNTPCLN4/5

You load the NLM via the LOAD SNTPCLN4/5 command. There are some commandline options available, but they are no longer supported and will be probably be removed in the next release. (V2.x)
Once the NLM is loaded, it checks for a the sntpclnt.cfg file in the SYS:ETC directory. It then reads the configuration. Once the parameters are loaded, the NLM looks for a file called sntpclnt.til in the SYS:ETC directory. If it is found, a previously saved tickrate is reloaded. Then normal operation begins.
Now every server listed in the cfg file is contacted to determine the current time. If the time difference is larger than the specified value in the configuration file, then a time correction is applied to the system clock. When the time corrections are small, then the correction is made via the tick interface. This means that the clock will never move backward. (A important requirement for NDS operation in multiserver environments)
When at least n time corrections have been made and the cumulated timecorrections are larger than specified in the cfg file, a new tickrate is calculated and used as the new tickrate.
The calculation of the new tickrate is based on the current corrections and the quality of the previous tickrates. So it will take some iterations until the tickrate will settle down around the „correct" value.

Before and after each server poll, the NLM checks if the configuration file has been updated. So you can modify parameters in the cfg file, and before the next poll occurs, the new parameters are loaded. This way you don't have to unload/reload the NLM, just to use new parameters.
Another thread is created as soon as the NLM is loaded. This one is responsible for managing the hardware clock. Every 3600 seconds it checks the time stored in the RTC with the current Netware time. If the difference is larger than 2 seconds, it tells Netware to write the current time to the RTC. This is important, so that when a server boots it directly has a accurate clock.

There are other threads active, but they are only used for the user interface and other less important tasks.

You can unload the NLM either via the keyboard, or via the UNLOAD command. When a poll is currently active, then the NLM first terminates it's TCP/IP activity before it unloads. This can take up to 10 seconds, depending the server timeouts specified in the cfg file.

Security considerations

When the Netware server is behind a firewall, then you will have to enable traffic in/out on the UDP port 123, otherwise the SNTPCLNT will not be able to contact the timeservers.
Since this is a UDP port, this is not a problem.

Configuration file SNTPCLNT.CFG

With the sntpclnt.cfg file you can change the behavior of sntpclnt to your needs. The CFG file is normally called sntpclnt.cfg and is located in the SYS:ETC directory. The configuration file is reread before and after each poll cycle. So you can make modifications to parameters and don't need to reload the NLM.
The configuration file has the same structure as normal INI files. It contains sections [Section] and entries. Please note that between the key and the = sign there must be a blank. The same is true after the = and before the value.
The default values are the one used by the NLM when it is loaded. If you override a parameter in the cfg file, then this value is used unless you specify a new value for this parameter. When you just remove a parameter from the cfg file, then the NLM does NOT set it's internal parameter to the default value, but rather continues to use the last specified value. (Until the NLM is reloaded)

Below is a list of possible sections and keys:

Section [GLOBAL]	General configuration
UseFirstAnswer = [**0**/1]	When set to 1, the servers are polled until a valid response is received. As soon as a valid answer is received, this time/offset is used, and the remaining servers are NOT contacted. This is interesting when you have a local NTP server available, but wish to have a backup NTP server, who can be contacted via a WAN in case of a failure of the local NTP server.
NtpSamples = **1** – 32786	The number of times a single NTP server should be contacted during each poll. When you set this to a value higher than 1, then the fastest (With the smallest round-trip time) is used. This option can be used to trigger a call to an ISP. Another possible usage is to give the routers/switches in your LAN time to relearn the correct route, when they have aged it out of the tables.
ShowClock = [**0**/1]	When set to 1, the current time is displayed in the upper right corner of the statistics screen.
AllowLargeCorrections = [**0**/1]	NW 4.x parameter. Normally SNTPCLNT only makes one large time correction after being loaded. All subsequent corrections are applied via the tick interface. If you wish to make some tests, where the clock should move forward/backward in larger amounts, then you can remove this tick-restriction. **Should NOT be used in productive environments.**
ResyncAT = xx:xx:xx	This entry can occur up to 24 times in a configuration file. With this entry you can specify exact times when to contact the NTP servers. So you can schedule NTP activity during off-peak hours, or when you have a internet connection.
LogFile = <logfile>	Where to protocol the messages.

SourceIP = <IP-number>	Specify which IP address to bind to the SNTP protocol. This is only important for multihomed servers, when you can't reach the ntp servers. Normally the IP stack determines the correct source address, so you don't have to use this entry.
ActivationKey = <key>	The activation key you got with your registration. (We no longer use the regkey utility, but you can still use it if you wish)
LogScreen = [**0**/1]	Display a logscreen who shows (almost) the same messages as the logfile.
MinTickRateCorrCount = 1 – 32768	Only used in NW 4.x. The minimum number of time corrections before a new tickrate is calculated. (Default value is 4)
MinTickRateCorrTime = 0.0 - 1.7E+308	Only used in NW 4.x. The minimum (absolute) cumulated time corrections needed, before a new tickrate is calculated. (Default is 1 second)
NTPOffset = +- 1.7E+308	The NTP offset to add to received times from the network. Can be used to compensate timezone problems. Default is 0.0
MaxCorrection = -1 – 32786	The largest possible correction to the local clock. If you wish to not restrict time corrections, set this value to –1. Default is 3700 seconds, so a daylight saving can be corrected if needed.
MaxTickCorrection = 1 – 32786	The maximum number of ticks to correct via the tick interface. Default is 300 seconds.
ResyncIntervall = 0 – 32786	Do a timepoll every n seconds. When set to 0, the ntp servers are contacted once, then the NLM unloads itself. (Unless you have specified ResyncAT entries)
MinimumCorrection = 0.0 – 1.7E+308	Only apply a timecorrection when the current time if off more than x seconds. For 3.x servers don't set this number below 1, for 4.x server you can go down to ~0.1 seconds. Default are 0.5 seconds.
MinimumHWCorrection = 0 – 4294967296	Only correct the CMOS (RTC) clock when the time difference exceeds x seconds. Default are 60 seconds.
HWCheckFrequency = 0.0 – 2147483	In which interval check the RTC clock for accuracy. Default are 3600 seconds.
ServerRetryCount = 1 – 32786	Try up to x times to contact a given timeserver during one time poll. Default are 5 times.
ServerTimeout = 1 – 32786	Wait up to x seconds for a answer from a time server. Default are 10 seconds.
LogLevel = 0 – 99	Verbose mode of logging messages. 0 = no messages, 99 = protocol every thing. (Gives large logfiles) Default is 3.
StatisticScreen = [0/1]	When set to 0, no statistic screen is displayed.

Section [TIMECORRECTIONS] Allows daylight saving for 3.x servers

Yyyy.mm.dd hh:mm:ss = 0 - 32786	Add x seconds to the NTP time after the given date, time. The date is always expressed in UTC (GMT)

Section [NTPServers] Timeservers to use

<server-ip-or-dns-name> [ntpversion]	You can specify up to 10 timeserver to contact for the correct time. When you have a DNS resolver installed on your system, then you can specify hostnames instead of IP addresses. For a list of public NTP servers look at http://www.eecis.udel.edu/~mills/ntp/servers.html When you don't specify the NTP version, the NLM will try to autodetect the version of the remote host. Only if this fails, you should put the versionnumber in the cfg file.

You can specify up to 10 (S)NTP servers . SNTPCLNT then contacts each server, calculates the "average" time of the servers, and then sets your server to this average time. It is a good practice to specify at least 3 time sources from the internet, so if you get some unusual transmission delays (as is normal in the internet on peak hours) then the other two servers will almost completely compensate this delayed time. (This is the minimum recommended number in the NTP protocol.)

With this you get two advantages:

1. When a (S)NTP server fails or is unavailable, then the chance is big, that another of the list can give you a correct answer.
2. When you have relatively high transmission delays (slow links, rush-hours etc.) then the received time is more accurate, since peak-loads on the net do not much influence your client.

Of course, you can specify the same (S)NTP server multiple times. It is the just asked multiple times for the current time, just as if it would be another server.

Year 2000 and other critical dates

Sntpclnt 1.1b, 1.5 and 1.6 are fully year 2000 compliant.
There are some interesting things to know about those subjects anyway:

1. Sntpclnt 1.1b displays the day of week in the logfiles/screens. You will notice, that on the currently shipping Netware 3.2 (and 3.12 with Y2K patches) the day of week shows a wrong value for dates after the 29.Fev.2000. This is due to a problem in the current CLIB.NLM (v3.12j), and is a confirmed Novell problem. They will provide an updated clib.NLM without this problem.
2. The Netware functions to set the hardware clock on 4.x servers seems actually to have a bug. When you switch the century, then the century information in the RTC is not updated. I have seen this on a IBM PS/2 server with Netware 4.11 and Y2k patches installed. I'm currently working on this with Novell. The 1.5 version of sntpcln4 has a work around for this bug. When after a hardware clock correction the century is incorrect, then NLM directly modifies the century register of the RTC.
3. The NTP protocol transmits the date/time as seconds since 1900. There are 32bits for the numeric representation of the seconds and another 32 bits for the fraction seconds. (Which gives 64bits per timestamp) The 32 bit can represent up to 4,294,967,296 seconds. Converted into date, this covers the period up to 7.Fev.2036. Accordingly to RFC 2030, an overflow will be considered as dates after 2036, so the NLM has a range from 1968 -–2104 (only considering NTP time stamps)
4. In the API and C libraries, there exists a time_t structure, use for various time/date calculations. This is a unsigned long and represents the number of seconds since 1970. Of course this field too will overflow at some time. This will occur somewhere in 2106. Novell (and some UNIX vendors) have announced to solve this issue in the next 64 bit OS they are currently working on.

Based on all the points mentioned above, we can say that the SNTPCLN4/5.NLM will work as expected in the range from 1990 – 2089. We are sure that this is enough for the currently available system.

Known problems / untested cases

Actually there are no known problems. Have a look at www.neatech.ch/sntpclnt for more information.

Future versions will...

Perhaps have the possibility to ping a host first, before trying to get the time. This would allow dial-on demand connections to establish a connection when needed.

What costs me all this ?

SNTPCLNT is shareware. You can get a activation key from the web site, who will activate the program for one month. You no longer use regkey.exe to apply the key to the NLM, but instead of this, put the key in the CFG file.

If you had a previous 1.x key, then you use it for all 1.x releases of sntpclnt. (No matter if Netware 3.x or 5.x)

SNTPCLNT is licensed on server basis. For each server you load SNTPCLNT you must obtain a license. When you buy a 3 – 5 server license, then you will receive one ActivationKey which can be used on all your servers. (up to 5)

Please have a look at www.neatech.ch/sntpclnt for the current registration information. Don't send check, since the fees to convert a check back into cash are ~10 US$.

The key you will receive works for all 1.x versions of SNTPCLNT. (No matter if sntpcln3 or sntpcln4)

	Price	www.shareit.com ID
1 – 2 Server license	15 US$	100353
3 – 5 Server license	20 US$	100352
6 – 10 Server license	30 US$	100351
11 – 25 Server license	40 US$	100350
26 – 50 Server license	50 US$	100349
51 - Server license	60 US$	100348

Where to get new versions

The first place to get the SNTPCLNT is on the internet at http://www.neatech.ch/sntpclnt/. With time different versions can be available at different other sites. But only the www.neatech.ch site will be up to date.

Technical considerations

Due to the nature of the (Netware) environment, there are a few things you should know about

- Normally you don't need to check you time very often. Once after server startup (in the AUTOEXEC.NCF would be a fine place) and then once day would be enough.

- NMX (NetBasic) support, currently not available. Does it make sense to implement this ? Tell me if you wish to see this in a future version of sntpclnt.

- You can specify hosts either as IP numbers or as DNS names. But for using DNS names, you must have a working DNS resolution installed on the server, in other words, netdb.NLM must be loaded. (Sorry, but resolv from hellsoft will not do it, since it's impossible to get any API) The 3.x version does not use DNS at all, only the etc/hosts file. There exists a netdb.NLM who can be loaded on 3.x servers, who would provide DNS capability, but it difficult to get the correct versions, so we have decided to NOT support DNS on 3.x servers.

- For a more detailed info about NTP look at this site http://www.eecis.udel.edu/~ntp

- There exists an older utility called RDATE in at least two different implementations. RDATE is based on RFC 868, SNTPCLNT is based on RFC 2030. Basically RDATE does the same as SNTPCLNT, but SNTPCLNT has a slight advantage, since it uses a better time protocol. (And you will find more servers who provide (S)NTP services than RDATE services)

Copyright

The beta versions of SNTPCLNT should not be redistributed in any way.

The final release versions can be distributed in any way, but the package must always include all files as in the original distribution.

You can distribute SNTPCLNT as part of your solution, but you have no right to sell it in any way to your customers. (Unless you buy a license for this client)

I do not guarantee that SNTPCLNT does behave in any reasonable way. You use this software at your own risk.

The legal stuff