BRAD KNOWLES

it's about time . . .

Brad has been using UNIX and the Internet for over 22 years, doing UNIX and Internet administration for over 16, and specializing in Internet email and DNS administration for more than a decade, and he now considers NTP and the NTP PSP to be his third principal area of specialty. He has spoken at a number of major conferences; was on the program committee for SANE 2000 and SANE 2002; was a reviewer for the second editions of *Sendmail* (O'Reilly, 1997), *DNS and BIND* (O'Reilly, 1997), and *Sendmail Performance Tuning* (Pearson Education, 2002); is currently involved in writing his own book; is co-authoring a booklet in the SAGE Short Topics series; and has been asked to be a reviewer of at least one other technical book in the field.

brad@stop.mail-abuse.org

AFTER 25 YEARS OF DEVELOPMENT [41], the Network Time Protocol (NTP) is now firmly established as the standard cross-platform way to set and maintain computer clocks on the Internet. Most modern OSes ship out-of-the-box with clients for NTP, and many of those are turned on by default. Most network devices have NTP clients built-in, and even many Small Office/Home Office (SoHo) DSL/cable modems/routers have them turned on by default. Unfortunately, as adoption spreads, misconfiguration is becoming more common, especially vendor misconfiguration. With misconfiguration comes bad or no clock synchronization and abuse or even "vandalism" of a surprisingly small number of time servers on the Internet.

The purpose of this article is to give you an update on the status of the protocol itself, the NTP Public Services Project (where you can get support for questions you may have regarding NTP), books and documentation related to NTP, the Top Five Most Common Problems, lists of publicly accessible time servers (including the NTP Server Pool project), time synchronization "state of practice" on the Internet, the release of updated "Reference Implementation" code, and recent developments on NTP server abuse (following David Malone's article from the April 2006 issue of *;login:*). Footnotes used will be in the asr (alt.sysadmin.recovery) tradition [32, 33].

NTP Working Group

The IETF is in the process of updating the NTP-related RFCs, specifically working toward an official specification for version 4 of the NTP protocol (RFC 1035 was published in 1992 and covered NTPv3).

Toward this end, they have set up an NTP Working Group (NTPWG) [14]. The mailing lists are being hosted [15] by the NTP Public Services Project [26], as well as a TWiki[40]. The NTPWG page [14] tells us:

> A number of topics have been raised as potential work items for an update to NTP including support for IPv6, security considerations including authentication, automatic

configuration including possible requirements for DHCP, and algorithm improvements.

If you're interested in helping to shape the future of the NTP protocol or the NTP implementations, please join the group and give us the benefit of your experience and views.

NTP Public Services Project

For years, the main site for most things related to NTP was at www.ntp.org. The ntp.org domain is owned by Dr. David Mills [34, 7], the Web site is maintained by his students at the University of Delaware and various members of the "volunteer corps," and the hardware is managed by the UDel staff. However, the support services eventually outgrew the hardware resources available at the host institution and, unfortunately, began to conflict with their policies.

Thanks to support from the Internet Systems Consortium, most of the NTP support services have been migrated to the NTP Public Services Project (NTP PSP) [26], as part of the "Hosted@ISC" programme, which includes Apache, FreeBSD, KDE, Mozilla, OpenBSD, OpenDarwin, OpenLDAP, OpenOffice, PostgreSQL, XFree86, kernel.org, and many others. Remaining at the ntp.org Web site are the main NTP home page, the NTP FAQ, the official download site for the source code, and tarballs of the "Reference Implementation," as well as continued research and development of the protocol and code by Dr. Mills and the "volunteer corps."

We are very grateful for all the assistance and hardware provided by ISC, and we'd like to thank all of our other donors as well [30]. A page for donating to the project has been set up [29], which includes information on how you can make tax-deductible donations through ISC and links to information on various pieces of hardware that we lack and hope to be able to obtain [31].

NTP Documentation

Dr. Mills is the person who originally created the NTP protocol and is sometimes called "Father Time" [7, 34]. He has recently published his book on the subject, *Computer Network Time Synchronization: The Network Time Protocol*, published by CRC Press (2006). Of course, it's already been reviewed on slashdot [8].

The only other dead-tree publication to cover this topic (so far) is *Expert Network Time Protocol: An Experience in Time with NTP* by Peter Rybaczyk (published by Apress, 2005, and reviewed by slashdot [9]).

For online documentation, there are the official pages written and maintained by Dr. Mills [10], the NTP FAQ [11], the Community Supported Documentation (CSD) [12], and many other pages linked from the NTP PSP Documentation page [13].

Top Five Most Common Problems with NTP (a.k.a. NTP Mini-FAQ)

One of the most common issues I've seen has been something along the lines of "I've done everything I'm supposed to, and it still doesn't work!" Here's a run-down of common causes:

1. They haven't punched a hole in their network firewall or host firewall software for bi-directional traffic on UDP port 123.

If you can't open port 123 for UDP in both directions, then you can't use the NTP daemon. The ntpdate program can be used with a "-u" option to tell it to bind to a high-numbered port, which may be allowed by the firewall configuration, but this sort of option is not (yet) supported by ntpd.

2. They have unknowingly configured the software to ignore all responses that are not cryptographically signed.
   Hint: The meaning of "notrust" changed between 4.1.x and 4.2.0. Disable "restrict notrust" unless you really understand what it's doing.
3. They may be running with SELinux enabled and not configured to allow the NTP software to update the system clock, etc.
4. They chose a set of upstream time servers that is not sufficient to allow the NTP algorithms to work correctly.
   Hint: Use either just one or at least three or more, because the person with two clocks never knows what time it is [35].
   In fact, you should use at least four or five upstream clocks if you want to be able to have one or more of them die or go insane, while your clock continues to function correctly. More information can be found in Section 5.3 of the CSD [16].
5. Their time zone is not correctly configured or is not properly displaying daylight savings time. The machine may be doing an adequate job in synchronizing the system clock to the upstream servers, but the presentation of this information is not correct.
   This is not an NTP problem, since NTP operates exclusively in Universal Coordinated Time (UTC). The conversion from UTC to the local time zone is considered to be a representation issue for the OS and is outside the control of the NTP programs.
   Make sure your time zone settings are correct in your /etc/localtime file, the $TZ environment variable, or otherwise as appropriate for your OS.

If you're having problems with NTP, we've got a whole section of the CSD devoted to troubleshooting [17] and describing common issues that people have, especially Section 9.1 on common hardware problems [18] and Section 9.2 on OS trouble [19]. If you've gone through all the documentation and you're still having problems, feel free to post on comp.protocols.time.ntp (which is gatewayed to the mailing list questions@ntp.isc.org [36]), or come see us on irc at #ntp on irc.freenode.net.

If you decide to use the irc channel, please be aware that there aren't many of us in the project and who monitor the channel on a regular basis, so you might need to wait a while for a response—perhaps several hours, or even a day or more. The mailing list/newsgroup is probably a better choice, unless you have a strong requirement for interactive support and you can afford to wait for it.

Also, if you see anything that could be improved in the CSD, or needs clarification, please feel free to sign up for a TWiki account and then dive right in to make the changes yourself. There's no way we can maintain all this information all by ourselves (in our nonexistent free time), which is why we created the CSD pages—so that everyone in the community would have the ability to contribute and correct information found there.

NTP Server Pool

If you are configuring your own NTP clients (or local NTP servers, from which your clients will be served), you should read the "Rules of

Engagement" [37], but you should also be aware of three different but related sets of public time servers. There is the list of public Stratum 1 time servers [20], which should only be used if you are setting up your own local NTP server(s) and are going to be serving local clients from it (them). There is the list of public Stratum 2 time servers [21], which can be used by individual clients as well as local time servers that will be redistributing time to their own local clients. Then there is the set of servers that comprise the "NTP Server Pool." See the NTP PSP Pool Servers page [22] and the NTP Server Pool Web site [23] for more information on how the pool works.

The purpose of the NTP Server Pool is not to give you the best possible time, but instead to help you fill out your list of upstream servers that should be able to give you a reasonable baseline, from which your client/server can pick out the best available source.

There are now over 600 public time servers currently available through the NTP Server Pool (out of more than 700 total defined in the database, about 100 of which are currently not being advertised owing to various problems), with about 400 (total) in Europe and over 200 (total) in the United States. However, there is still a desperate need for additional time servers in the pool for other zones.

As of 24 April 2006, Ask Bjørn Hansen (the current maintainer of the NTP Server Pool) estimates that there are somewhere between two and six million client systems that are using the pool. You can help the project stay alive by contributing to the pool, if you have a static IP address [24].

Time Synchronization State of Practice

A question came up on the sage-members mailing list about the state of practice of time synchronization, and wondering why this doesn't seem to be more universally deployed at the server and client level.

I can't speak for the operational practices for most organizations, but I can say that more and more vendors are enabling NTP or SNTP code out-of-the-box. With recent versions of Windows, Microsoft ships an SNTP client, and they provide their own time servers for those clients to connect to. Apple has provided an NTP client in MacOS X for quite some time, making it easy to enable and configure and also providing time servers for those clients to connect to.

FreeBSD, NetBSD, and most other *BSD implementations not only ship NTP clients out-of-the-box, but they also enable them by default. Many Linux distributions are doing the same.

For vendors that configure their clients to use NTP by default, the practice within the community has been to encourage those vendors to also supply some time servers for those clients to use, or to configure their DNS in a particular way to allow them to make use of the servers provided through the NTP Server Pool project in a way that will minimize negative impact [38].

However, not all free/libre/open-source systems (FLOSS) platforms have felt that they have the ability to provide servers directly. Instead, some FLOSS platforms are actively encouraging their members to help provide additional machines for the NTP Server Pool. Debian is probably the best known in this regard, but Red Hat is providing their own Stratum 1 and Stratum 2 servers (see the aforementioned lists), as well as listing these machines in the pool.

Poul-Henning Kamp (from the FreeBSD project) runs a couple of restricted-access Stratum 1 time servers, and thanks to donations of GPS reference clock hardware from Meinberg, the NTP PSP also hopes to make available at least two Stratum 1 time servers of their own.

In addition, more vendors are shipping embedded hardware with NTP enabled, even though some of them make mistakes and misconfigure the firmware in their devices.

Most dedicated network devices (especially routers) come with NTP clients built in and can even act as NTP servers (although this may not be a good idea [25]; see "Sidebar," p. 30).

At this point, the only observation I can make is that we must tend to get one of two situations:

1. Many people apparently configure this stuff and do so with relative ease and don't feel the need to tell anyone. Thus we don't hear about the positive cases.
2. Many other people probably still don't see the need to have good time sync on their machines. Thus we don't even know about the cases where we never even got considered.

But there were a surprising number of people on the sage-members list who spoke up and said that, based on their personal experience, network-wide NTP time synchronization was a much more common thing than you (or I) might think.

Updated Code

As of the time of this writing, the NTP PSP has recently released version 4.2.2 of the "Reference Implementation" of the NTP protocol [27]. By the time you see this article, we hope that many vendors will already have picked up this greatly improved code and incorporated it into the software they are shipping.

NTP 4.2.0 was released on 15 Oct 2003. Version 4.2.1 has been in development since, with many improvements made over the years. Unfortunately, many vendors have stuck with the "stable" 4.2.0 codebase, instead of tracking the improvements that have been made in the 4.2.1 development tree.

This has left many in the community with various known bugs and weaknesses that have already been fixed in the source tree, and they have found themselves in the uncomfortable position of either having to remove the vendor-provided code and replace that with code based on the source tarballs available from the NTP PSP download page [27] or waiting for someone to create a binary packaged version for them to download and install. Both approaches cause configuration management problems.

With the advent of version 4.2.2, we're going to be changing our release numbering scheme slightly [28], and we hope to be able to release new versions much more frequently than every few years. For now, we're targeting at least two new releases per year.

We are now also creating cryptographic hashes for the source tarballs, and we hope to start PGP-signing the announcements so that you can be reasonably sure that the code you're downloading is actually the code we released.

You may be interested to know that we also provide an RSS 2.0 feed of our current tarball information [39].

Sidebar: Time Server Abuse

In the April 2006 issue of *;login:* you may have read David Malone's article "Unwanted HTTP: Who Has The Time?" [42]. To summarize: there were thousands of clients worldwide running a program called Tardis, connecting to his server and obtaining a timestamp via HTTP. These clients were connecting as frequently as once an hour or even once a minute. Traffic volume was estimated at 30 GB/month, based on the initial data collected after enabling increased logging.

Although these clients were connecting via HTTP, this is a classic case of time server abuse by misconfigured clients. Unfortunately, it's not the only case, or even the most recent one.

A better-known case is found at the University of Wisconsin [1], where:

> [They were] the recipient of a continuous large scale flood of inbound Internet traffic destined for one of the campus' public Network Time Protocol (NTP) servers. The flood traffic rate was hundreds-of-thousands of packets-per-second, and hundreds of megabits-per-second.

Ultimately, all this traffic was discovered to be the fault of misconfigured NetGear cable/DSL routers with embedded IP addresses as their set of predefined (and nonoverrideable) NTP time servers. At least NetGear was willing to work with UWisc and Dave Plonka to try to resolve the problem as well as possible, and the company has made a donation to the university for their help in locating and helping to get this problem fixed [2].

Just after the April 2006 issue of *;login:* came out, another instance of time server abuse came to the forefront. This time, it was the time server run by Poul-Henning Kamp at the Danish Internet Exchange, for the benefit of network providers in Denmark and their customers. Again, the fault lay with a commercial product with a bad default configuration (in this case, D-Link cable/DSL routers). However, this time the company took the notice by Poul-Henning to be an act of extortion, sending their lawyers after him.

The issue is now supposedly settled [3], so Poul-Henning has taken down the original notice, but you can still read about the story on other Web sites [4, 5, 6].

Wikipedia also has a good page on the subject of time server abuse [6], including a reference to a similar abuse problem that occurred between SMC and the CSIRO in Australia.

When all is said and done, one question you have to ask yourself is whether or not you want to be using hardware from a company that acknowledges the problems that they may accidentally create for others and works with you to try to resolve them.

What happens when you're on the other end of that pointy stick and your servers are being nuked off the Internet? What kind of response do you want to see from the company that is responsible?

REFERENCES

[1] http://www.cs.wisc.edu/~plonka/netgear-sntp/.

[2] http://www.doit.wisc.edu/news/story.asp?filename=322.

[3] http://people.freebsd.org/~phk/dlink/.

[4] http://yro.slashdot.org/article.pl?sid=06/04/07/130209.

[5] http://www.lightbluetouchpaper.org/2006/04/07/when-firmware-attacks-ddos-by-d-link/.

[6] http://en.wikipedia.org/wiki/NTP_vandalism.

[7] http://www.udel.edu/PR/Messenger/02/1/where.html.

[8] http://books.slashdot.org/article.pl?sid=06/05/15/143251.

[9] http://books.slashdot.org/article.pl?sid=05/08/16/0344212.

[10] http://www.eecis.udel.edu/~mills/ntp/html/index.html.

[11] http://www.ntp.org/ntpfaq/NTP-a-faq.htm.

[12] http://ntp.isc.org/support.

[13] http://ntp.isc.org/doc.

[14] http://www.ietf.org/html.charters/ntp-charter.html.

[15] https://lists.ntp.isc.org/mailman/listinfo/ntpwg.

[16] http://ntp.isc.org/bin/view/Support/SelectingOffsiteNTPServers.

[17] http://ntp.isc.org/bin/view/Support/TroubleshootingNTP.

[18] http://ntp.isc.org/bin/view/Support/KnownHardwareIssues.

[19] http://ntp.isc.org/bin/view/Support/KnownOsIssues.

[20] http://ntp.isc.org/s1.

[21] http://ntp.isc.org/s2.

[22] http://ntp.isc.org/pool.

[23] http://www.pool.ntp.org/.

[24] http://www.pool.ntp.org/join.html.

[25] http://ntp.isc.org/bin/view/Support/DesigningYourNTPNetwork#Section_5.6.

[26] http://ntp.isc.org/.

[27] http://ntp.isc.org/download.

[28] http://ntp.isc.org/bin/view/Main/ReleaseNumberingScheme.

[29] http://ntp.isc.org/bin/view/Main/DonatingToTheProject.

[30] http://ntp.isc.org/bin/view/Main/OurDonors.

[31] http://ntp.isc.org/donat.

[32] news:alt.sysadmin.recovery.

[33] http://www.faqs.org/faqs/sysadmin-recovery/index.html.

[34] http://www.eecis.udel.edu/~mills/bio.html.

[35] http://www.quotationspage.com/quotes/Segal's_Law.

[36] https://lists.ntp.isc.org/mailman/listinfo/questions.

[37] http://ntp.isc.org/bin/view/Servers/RulesOfEngagement.

[38] http://www.pool.ntp.org/vendors.html.

[39] http://ntp.isc.org/rss/releases.xml.

[40] http://ntp.isc.org/ietf.

[41] http://www.eecis.udel.edu/~mills/database/papers/history.pdf.

[42] http://www.usenix.org/publications/login/2006-04/pdfs/malone.pdf.