FDD | CCTI Center on Cyber and Technology Innovation

Research memo

How Many Seconds in a Second? Why Keeping Time on Computers Is Hard and Why It Matters

Michael Hsieh, Ph.D., and Douglas Wood, Ph.D.
Transformative Cyber Innovation Lab

July 22, 2020

Note: This technical report was prepared for Project ICARUS, which TCIL launched in response to the Cyberspace Solarium Commission report's Recommendation 4.5.2, which focuses on improving foundational internet protocols.

Why is it important to keep time accurately on computer networks? An accurate time standard is a basic requirement of operating a computer network.[1] Keeping an accurate time-ordering of events is necessary for distributing tasks across many computers on a network. If events are not ordered correctly, a computer network can malfunction or crash. Any cybersecurity stakeholder with a policy or operational responsibility for network resiliency must appreciate the importance of the foundational protocols on which all networks run, which certainly include timekeeping.

With the way time is kept on computer networks today, there are many ways timekeeping can go wrong because of (a) the physics of the geographical distance between computers in a network, and (b) the complex behaviors that emerge in networks when disparate protocols are cobbled together. To demonstrate just how easily such problems can arise, even in simple systems, the authors built some small-scale computer networks to use as a testbed. The results suggest several recommendations for improving practical timekeeping on networks, as well as imperatives for policymakers thinking about how to secure foundational internet infrastructure more generally.

How well do computers keep time? Digital computers are synonymous in the popular mind with precision processing of information. However, when it comes to the basic information processing task of keeping time, they often perform worse than expected. To see how bad computers can be, consider some things that are better at keeping time. The gold standard in timekeeping is the sundial, which is a perfect analog timekeeper.[2] However,

1. Leslie Lamport gave the classic statement of the time-ordering problem in distributed computing: Leslie Lamport, "Time, Clocks, and the Ordering of Events in a Distributed System," *Massachusetts Computer Associates, Inc.*, July 1978, Vol. 21, No. 7, pages 558–565. (https://lamport.azurewebsites.net/pubs/time-clocks.pdf)

2. For an overview of modern sundials, see: "A Sundial as an Accurate Time-piece: A Spot of Light That Tells the Time of Day," *Scientific American*, July 29, 1911. (https://www.scientificamerican.com/article/a-sun-dial-as-an-accurate-time-piec/)

Douglas Wood is the chief scientist at Ursa Space Systems and a member of the Board of Advisors of FDD's Center on Cyber and Technology Innovation. He previously led architecture development for extremely large distributed computing systems as a senior technical leader at the Defense Information Systems Agency. Michael Hsieh is the executive director of FDD's Transformative Cyber Innovation Lab.

sundials are not very practical and are not used to keep time on computer networks. The U.S. government, for instance, often needs to run its computer networks at night or on cloudy days.

There are good alternatives. Global Positioning System (GPS) time standards have an accuracy better than 40 nanoseconds (billionths of a second) 95 percent of the time.[3] And how do computers fare against GPS? Not well. As a rule, computers can err by up to a few minutes per day. Examining how networked computers keep time will demonstrate just how easily such bad performance can occur.

How do networked computers keep time? Networked computers keep time using what is called Network Time Protocol (NTP).[4] In a computer network, some computers may need to know the time. These are called "client" computers. Then there are other computers that keep time. These are called "server" computers, or more precisely, NTP servers.

To define poor timekeeping performance, one must first define what it means for a timekeeping system to work well. The clocks on computer networks are not just any clocks. Rather, these clocks are used to schedule complex computer operations across a network that will either malfunction or crash if any computer's timekeeping function acts too oddly. A good network timekeeping system should obey at least two rules: First, time should always be moving forward. Second, if the time must be corrected, this correction should happen smoothly.

In computer networks in the real world, timekeeping frequently falls short of this standard due to (a) the physics of geographical distance and (b) the complexity of the systems involved. The next section demonstrates these points.

How can computer network timekeeping can go wrong? To measure what kinds of errors are caused by distance, the authors set up one computer with a commercial operating system (OS) and another computer with an open-source OS. The two computers communicate with an NTP server separately and independently. The authors issued NTP requests from both computers from the Washington, DC, metro area and recorded the roundtrip time to and from an NTP server located outside of the Washington, DC, metro area. [FIGURE 1]. The roundtrip times were about the same for both computers.

Next, the authors examined the kinds of clock errors caused by these latencies. Each OS has a clock-correction algorithm designed to correct for such latencies, but depending on the particulars of how the algorithm is implemented on each OS, the actual clock errors can vary. In the case of the computer running the commercial OS, the errors are serious: It walked off the correct time by about a minute and was actively worsening over time [FIGURE 2]. By contrast, the errors of the computer running the open-source OS were not even perceptible on the same scale – on the order of hundredths of a second.

Next, the authors demonstrated how errors arise when communications between NTP servers and clients get confused. To show how this can happen, the authors put two computers running on the same commercial OS behind the same proxy firewall. A proxy firewall protects computers from outside threats by acting as a single gateway for all internet traffic coming in and out. Outside computers do not talk directly to computers inside the

3. The U.S. Naval Observatory maintains the time standard used by GPS. U.S. National Coordination Office for Space-Based Positioning, Navigation, and Timing, "GPS Accuracy," accessed July 21, 2020. (https://www.gps.gov/systems/gps/performance/accuracy/#how-accurate)

4. The community of volunteers and researchers supporting NTP development is several decades old. "NTP: The Network Time Protocol," *NTP Project*, accessed July 21, 2020. (http://www.ntp.org)

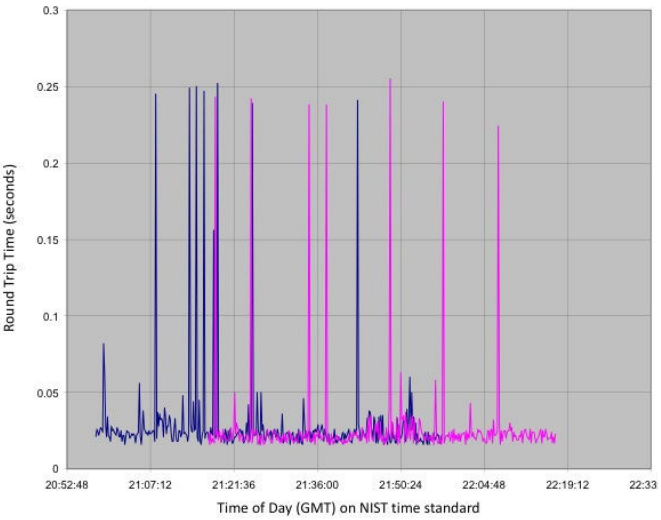

Figure 1

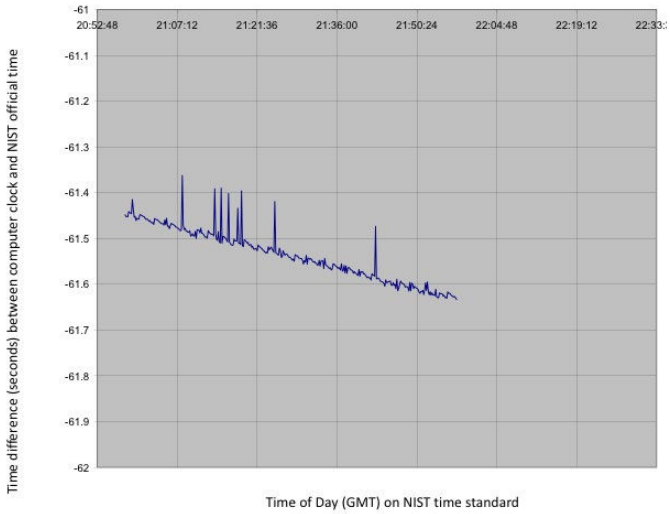

Figure 2

firewall; they talk to the firewall, which replicates their messages after checking them for security issues and then sends the messages to the computers inside the firewall. When the computers inside the firewall want to talk to the outside world, they also talk to the firewall, which replicates their messages to the outside.

While such firewalls add a high degree of security, that security comes at the cost of introducing the potential for confusion in communications. For example, if multiple computers are making outbound requests to the same outside NTP server, how does the inbound NTP server traffic return information to the right computer when all communications are funneled back into a single firewall gateway?

Fortunately, there are computer protocols and methods designed to solve this problem. Two commonly used ones are Network Address Translation (NAT)[5] and Session Traversal Utilities for NAT (STUN),[6] which are designed to make sure that such internet traffic is routed correctly in these situations. Unfortunately, the concatenation of such basic protocols, even on a simple network, sometimes produces confusion in the firewall that causes traffic to be sent to the wrong place, resulting in serious clock errors.[7]

In this demonstration, the authors found that the clock times on both computers began to deviate in serious ways when NTP time updates went to the wrong computer. The authors compared the clock errors on both computers behind the firewall to an independent time standard synchronized to the National Institute of Standards and Technology national time standard. For one computer, the error worsened smoothly, albeit continually, over the

5. As a rough analogy, NAT loosely corresponds to the division of mail among multiple inboxes and outboxes in a household of several persons who share a single physical mailbox. For a detailed explanation of NAT, see: "Network Address Translation (NAT) FAQ," *Cisco*, accessed July 21, 2020. (https://www.cisco.com/c/en/us/support/docs/ip/network-address-translation-nat/26704-nat-faq-00.html)

6. Continuing with the foregoing analogy for NAT, STUN loosely corresponds to the household's process of sorting incoming mail when outside parties write only the address and not the names of the specific recipients for which the mail was intended. This relates to the sessionless aspect of NTP server traffic, which is generally based on User Datagram Protocol instead of on a sessioned protocol such as Transmission Control Protocol (TCP). This report does not go into any deeper level of detail on this matter, but interested readers can explore the many good open-source resources on STUN. See, for example: J. Rosenberg, R. Mahy, P. Matthews, and D. Wing, "Session Traversal Utilities for NAT (STUN)," *Internet Engineering Task Force*, October 2008. (https://tools.ietf.org/html/rfc5389)

7. NTP is old enough that it does not use STUN but requires that firewalls that implement NAT attempt to keep track of which particular requesting computer behind the firewall is supposed to receive a particular NTP response.

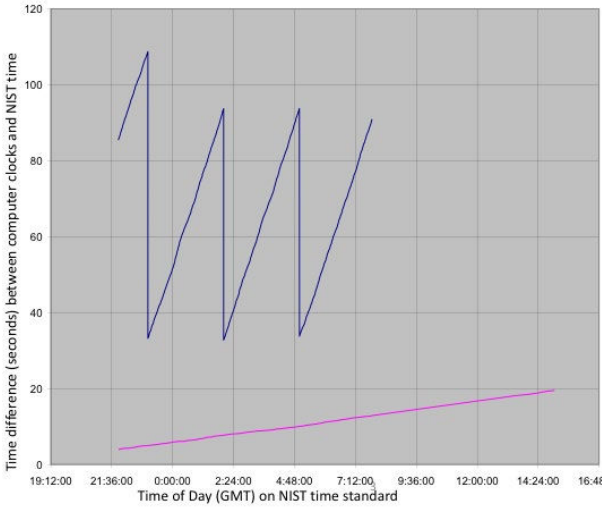

Figure 3

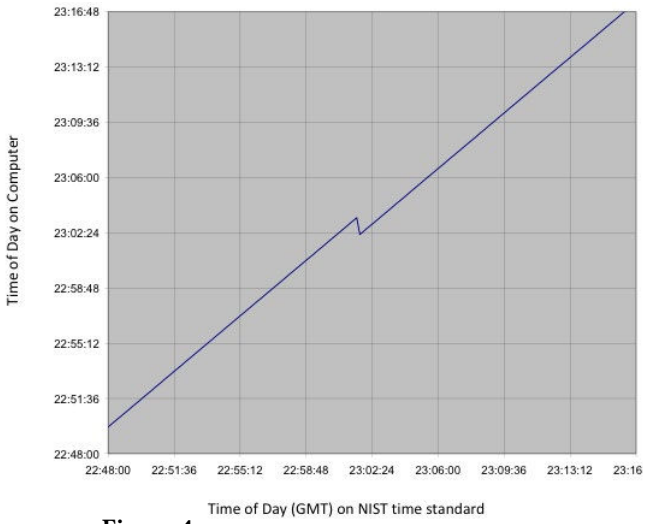

Figure 4

course of about a day. The error was approximately 20 seconds. For the other computer, the error was approximately 40 to 100 seconds over the same time period, with unsmooth corrections every few hours [FIGURE 3]. For one computer, the authors even observed an episode during which the computer broke one of the basic rules for a well-functioning timekeeping system: For an 80-second interval, the clock time actually moved backward [FIGURE 4].

How can timekeeping on computer networks be improved? The sort of problems described here have solutions that are implementable at the enterprise level of policy.

For those enterprises with the resources to do so, GPS-based timekeeping is the most widely adopted resource for the time-ordering of financial and commercial transactions, both among businesses around the world as well as among operators of physical infrastructure spread over large geographical distances.[8] The wide adoption of GPS as a time standard underscores the vital national importance of GPS and the imperative for policymakers to invest properly in (a) securing GPS and (b) developing viable alternative national-scale time-standard assets in preparation for long periods of GPS unavailability (because of malice or otherwise).[9]

For computer networks that *must* use NTP as their time standard, this experiment's results suggest that NTP can get confused when multiple computers are sharing the same routable internet protocol address, as is the case when they are behind common network assets such as firewalls. So, within such a computer network, only one computer should be designated as the timekeeper, and all NTP traffic should be directed specifically to that computer and that computer alone. While such network architecture specifications seem commonsensical, they are not universally adopted.[10] Insofar as NTP must be used by some enterprises and industries, a baseline set of proper NTP configuration standards should be developed and adopted.

8. U.S. National Coordination Office for Space-Based Positioning, Navigation, and Timing, "Timing," accessed July 21, 2020. (https://www.gps.gov/applications/timing/)

9. There have been decades of efforts invested in alternatives to GPS to prepare for a catastrophic GPS-failure situation. See: F. Pappalardi, S.J. Dunham, M.E. LeBlang, T.E. Jones, J. Bangert, and G. Kaplan, "Alternatives to GPS," *Institute of Electrical and Electronics Engineers*, 2001, Vol. 3, pages 1452–1459. (https://ieeexplore.ieee.org/document/968047)

10. In the authors' experience, systems administrators sometimes reset network clocks according to the time shown on their wristwatches. Time-standard practices vary greatly across networks and enterprises and are generally less uniform than often imagined.