# The Importance of Network Time Synchronization for Enterprise Solutions

**Whitepaper**

Microsemi
Power Matters.™

**Microsemi Corporate Headquarters**
One Enterprise, Aliso Viejo,
CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
E-mail: sales.support@microsemi.com
www.microsemi.com





## About Microsemi

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 4,800 employees globally. Learn more at www.microsemi.com.

# Timekeeping Requires More than Just Great Timing

As you read this, your network of workstations and servers, each with its own clock, are actively timestamping files, emails, transactions, and so on. At the same time, your server logs are recording every type of transaction in case the information is needed for audit or forensic analysis. At some point during the day, it is quite likely that automatic processes (such as archiving, directory synchronization, and cron jobs) will execute and alter files based on these timestamps.

Fundamental to all this is the belief that the time is correct. But is it? To answer that question, you must first consider:

- Is the time source accurate and precise?
- Is it secure? That is, is the time itself vulnerable or does it make the network vulnerable?
- Is timing easy to use and deploy across the network?

In many of today's networks, the answer to the first question is, “maybe not.” Computer clocks, for instance are notorious for drifting. They are typically based on inexpensive oscillator circuits or battery backed quartz crystals that can easily drift seconds and minutes per day, accumulating significant errors over time.

That is why most enterprise networks today rely on network time servers that acquire time from the Global Navigation Satellite System (GNSS) and distribute it to clients over a network through the Network Time Protocol (NTP). These NTP servers also employ stable internal oscillators in case the server loses the satellite signal. Oscillator holdover accuracy varies based on the type of oscillator: from 400 microseconds over the first 24 hours of signal loss for a standard quartz oscillator to less than 1 microsecond for a rubidium oscillator.

Timing accuracy from NTP servers depends upon the accuracy of the source and the precision with which all operations and applications are synchronized. However, accuracy and precision may not matter if the NTP server is vulnerable to attack or if it exposes other parts of the network to attack. There are, however, a number of value-added measures that can be taken against threats like unauthorized access or Denial of Service (DoS) attacks where a flood of packets can bring down the NTP server. Quality time is also better served if network administrators find the timing solution easy-to-deploy and use. The major questions that arise when quality time has to be served are:

- How easy is it to support various networking topologies, workloads, and security schemes (for example, TACACS+, RADIUS, and LDAP user access authentication)?
- Is the user interface web based?
- Is it logically organized?
- Are controls and displays intuitive and easy to master?

The quality of time on a network is based on the quality of time at the time server. It ensures reliably accurate and precise time, and is the starting point for considering all the methods by which an NTP server can add value beyond timekeeping.

# A Pervasive Need for Quality Network Timing

Quality time in an enterprise network is critical to operate the network in a reliable and secure manner and support the applications. It is also essential to ensure the performance of the network and assure compliance and forensics. Network operations require time-synchronized information to ensure optimal network performance. It often is not until a problem occurs that organizations become aware of the importance of time synchronization (either as a contributing factor to the problem itself, or as a necessary tool to diagnose the problem.) Many network processes will not function at all without proper time synchronization.

Likewise, many applications rely heavily on access to accurate and precise timing to perform a wide range of functions. For example, many applications use timestamps as a key element that adds tremendous meaning to measured and generated data. Shared databases, billing and transaction systems, data acquisition, email, and many more applications rely heavily on accurate timestamps of varying degrees of precision. With the increasing scope of timestamp usage, many common applications rely on network synchronization to provide the time for a meaningful timestamp.

Table 1 lists the key areas where quality network timekeeping is required in both network operations and applications. The timing solution's security, ease of deployment, and ease of use also matter in these areas.

***Table 1 •*** **Key Areas Requiring Quality Network Timekeeping**

| Network Operations | Applications |
|---|---|
| Log file accuracy, auditing, and monitoring | Transaction processing |
| Network fault diagnosis and recovery | Distributed processing |
| File timestamps | Software development |
| Virtual environments | Email |
| Directory services | Legal and regulatory requirements |
| Access security and authentication | |
| Scheduled operations | |
| Real-world time values | |

These are also areas where the timing solution's security, ease-of-deployment, and ease-of-use matter. Table 2 lists the security features that can be included in an NTP server.

***Table 2 •*** **NTP Server Security Features**

| Feature | Function | Benefits |
|---|---|---|
| NTP Reflector™ with hardware timestamping | 100% hardware-based NTP timestamping operations | • Extremely high-accuracy and high-throughput NTP operation<br>• Ensures DoS attacks won't bring down the NTP server<br>• Detects DoS attacks so system administrator can be notified<br>• Bandwidth limits non-NTP traffic<br>• Enables single server support for very large networks |
| CPU-based bandwidth limiting | Allows only a predetermined number of packets to reach the NTP server CPU | Protects against DoS attacks |
| HTTPS/SSL | Encrypts management traffic between server and its web interface | Protects against data theft and unauthorized server entry control |

***Table 2 •*** **NTP Server Security Features (continued)**

| Feature | Function | Benefits |
|---|---|---|
| Password access | Requires a password to manage the server | Ensures authorized-only administrative access |
| NTP MD5 authentication and NTP autokey | Provides key-based hashed NTP packet exchange between clients and servers | Ensures NTP packets cannot be spoofed as a way to corrupt the time |
| Access control lists | Limits clients that can access the server to an administrator-specified list | Protects against unauthorized server use or entry as an attack vector of the network |
| TACACS+, RADIUS, and LDAP authentication | Limits who has management access to the server using credentials based on industry-leading access management systems | Enables the server to be managed as part of a network-wide access management system (administrators do not have to manage credentials locally for each device). |

Table 3 lists features that help ease of deployment and use.

***Table 3 •*** **NTP Server Deployment and Use Features**

| Feature | Function | Benefits |
|---|---|---|
| Web-based Management | Provides at-a-glance displays and logically organized controls | Makes it more likely that timing and security settings will be optimal for the particular network on which the server is deployed |
| Multiple GbE ports | Provides port-independent NTP timestamping as though multiple clocks serve different subnets | Provides flexibility to adapt to different network topologies as networks grow and change |
| NTP hardware-based timestamping | Uses dedicated hardware to accurately timestamp NTP packets | Improves the time stamp accuracy at all NTP request levels |
| Dual power supplies | Provides uninterrupted backup should a power supply fail | Allows for reliable, unattended operation even in remote locations or during non-working hours |
| Upgrades to OCXO and Rubidium oscillators | Provides <25 microseconds (OCXO) or <1 microsecond (Rubidium) holdover drift over 24 hours | Maintains accurate, precise timekeeping in the event that GNSS satellite reception is lost due to terrain, urban canyons, jamming, or other reasons |
| Support for multiple GNSS constellations | Enables the server to acquire time from different nations' GNSS satellites | In the event primary GNSS satellite reception is lost, accurate and precise timekeeping can be maintained |
| Extended environmental specifications | Allows the server to be stored and operated under extreme conditions | Enables the server to be more widely deployed in harsher environments with less climate control |
| TACACS+, RADIUS, and LDAP authentication | Limits who can access the server based on whether or not they have credentials listed in one of the mentioned industry network authentication schemes | Supporting multiple industry-standard authentication protocols, allowing the server to be easily deployed and managed on more types of networks |

These two sets of features support timekeeping by enhancing security and availability in various operational settings and application environments. On the network operations side, there is no better example than achieving log file accuracy.

# Log File Accuracy, Auditing, and Monitoring

Server log files and subsequent reports enable assessment of network activities. This includes firewall and VPN security-related activity, bandwidth usage, various logging, management, authentication, authorization, and accounting functions. Because server logs are a compilation of information from different hosts, accurate timestamps are essential for ordering events, and identifying and troubleshooting root-causes. Statistics on time related factors are difficult to interpret and possibly meaningless without accurate timestamps. Even in routers, centrally logged configuration events and system error messages (such as router configuration changes, interface up/down status, security alerts, environmental conditions, trace backs, and CPU process overloads) rely on network time synchronization for accurate timestamps so that the data has meaning.

A number of enterprises have suffered from highly-publicized DoS attacks and data theft. In such cases, network security experts use log files (typically used to detect root-cause network events) to reconstruct the scene of a network crime. Accurately timestamped network packet transits provide the forensic evidence to make this possible.

NTP server security and deployment typically enhances the ability to achieve better log file accuracy (or to satisfy any other timing requirement). Some of the features worth noting are oscillator upgrades and features that defend against DoS attacks (such as hardware timestamping and CPU-based bandwidth limiting).

# Network Fault Diagnosis and Recovery

Most IT organizations are measured on their ability to maintain full flow network operations. Strict limits on allowable downtime are some of the most common quality of service (QoS) metrics in place, and every IT department is acutely aware of them. In the event of a failure, accurate network timing is crucial for fault diagnosis and recovery.

To assist in fault diagnosis, loss of connection, buffer over-flow, and so on, key network events are trapped, reported, and logged (typically using syslog services that reside in servers, routers, switches, and dedicated instruments). Should the network fail, a root-cause analysis is initiated, looking through the reported stream of events. Each of these events is indexed with the network timestamp affixed by the reporting agent. If these timestamps are synchronized, the proper order can be established and the root-cause quickly identified. Root-cause isolation is obscured and downtime is prolonged without accurate network synchronization.

Great timing is just the start…

If you are using your time server to help maintain network QoS, then the measures that enhance the time server's own QoS are critically important. These include features such as peering and holdover (in case the GNSS reference is lost), support for multiple GNSS constellations (if the primary GNSS reference is lost), dual power supplies (in case one power supply fails), and measures against DoS attacks. Another important feature is SNMP support that allows the sever to communicate an out-of-limits condition to administrators before the condition leads to a server or network failure.

# File Timestamps

The integrity of any file system relies heavily on the name and dates of the files. Individual files typically track the dates for creation, last access, last archive, and last modification. In a distributed file sharing system, a master file is maintained by a Network File Sharing (NFS) server for use by remote clients. NFS is network time-dependent: when presented with duplicate file names, it saves the latest copy. However, if a client timestamps a remotely-accessed file with a time earlier than the file maintained on the server, the client file, along with any changes, will be discarded.

Great timing is just the start…

If networks are large, composed of multiple subnets, or serve hundreds or thousands of clients, then timestamp accuracy across the network depends not just on the accuracy of the time reference at the server, but also on the ability to communicate the time efficiently (that is minimizing the time delay of actually applying and distributing the timestamp). That calls for features like hardware timestamping that apply timestamps at line speed, and can synchronize clients on multiple isolated subnets as if each subnet has its own dedicated time reference.

# Virtual Environments

Within virtual environments such as VMware and Hyper-V, guest virtual machines (VMs) share a host's physical resources, which affect the guests' ability to keep their own accurate time. Accurate time in VMs depends on regular servicing of timers so that the VM clock continually moves forward smoothly. A VM must often wait while the host OS is busy servicing other guests, typically resulting in the VMs' times falling behind or exhibiting erratic time behavior due to missed clock ticks. As with a real machine, the integrity of all operations of a virtual machine depends on keeping the time up-to-date (in other words, on not relying exclusively on the host resources).

Great timing is just the start…

A dedicated NTP time server coupled with good NTP clients can work to offset the negative timekeeping effects prevalent in virtual environments. While the guest VM is subjected to irregular clock ticks from the host, good NTP client software running on the guest can compensate for and work to overcome those timing irregularities.

# Directory Services

Network time synchronization is an important part of network design and implementation. For example, many network directory services systems exchange information and synchronize changes in the directory services database according to timestamps. Groupware applications require accurate time for scheduling and collaboration. Without a time-synchronized network, time-sensitive systems and applications will not work correctly. In a Windows active directory network, all Primary Domain Controller (PDCs) and client workstations need to synchronize with a single, accurate, and standard time source.

Great timing is just the start…

If one of the key roles of a network time server is to support directory services, then for purposes of management convenience and efficiency, it would be helpful to have the time server itself as one of the resources actually managed through the directory service. After all, managing resources from a central directory is why you would have the directory service in the first place, and a server that supports multiple directory services (such as LDAP) will more likely support the directory service on your particular network.

# Access Security and Authentication

Windows has long been a prominent example of the requirement for network synchronization. Synchronized time is critical in Windows because the default authentication protocol (Kerberos) uses workstation time as part of the authentication ticket generation process. Windows includes the W32Time Service tool to ensure that all Windows-based computers in an organization use a common time. The time service uses a hierarchical relationship that controls authority and does not permit loops so as to ensure appropriate common time usage. This continues down through the hierarchy of domains to the PDC at the root of the tree. This PDC is set to synchronize with a reliable time source, such as a dedicated network time server. If a time server is not available and the time difference between domain controllers drifts beyond the skew allowed by Kerberos, authentication/logon between domain controllers and clients may not succeed, and error messages can result.

Great timing is just the start…

For domain controllers to not drift out of sync, the time server needs to stay online and so does its reference to a good time source (such as GNSS or a very stable holdover clock). Critical features that support always on availability of accurate and precise time include oscillator upgrade options (to increase holdover precision if GNSS is lost), peering (in case GNSS is lost), support for multiple GNSS constellations (if primary GNSS is lost), and dual power supplies (if primary power supply fails).

# Scheduled Operations

Cron scripts and crontabs are a list of one or more commands to a computer operating system or application server that are to be executed at a specified time. Each command is executed when its triggering time arrives. These commands are commonly data backup oriented, and happen at pre-specified times that are intentionally scheduled late at night or after the close of business. Synchronization of a single host with an acceptable time source is mandatory so that commands run when expected. In the case of multiple hosts responsible for executing independent cron files, time synchronization between the hosts becomes even more critical to ensure that scheduled activities are properly coordinated.

Great timing is just the start…

The reliability features just listed would also assure that scheduled operations are executed as per expectations especially during unattended hours.

# Real-World Time Values

There is no substitute for operating a network using real-world time values. While you can synchronize a network to the incorrect time and make it work, this is a very undesirable policy. Local networks are interconnected with other larger networks, particularly over the Internet, and correct time is the single common denominator. Real-world time is based on Universal Time Coordinated (UTC). Networks operating on the underlying UTC share a common time base. UTC time is best obtained from an accurate, secure, and reliable source, and then converted to local time by all operating systems that reference that source. It is this common time reference that provides network managers the time-accurate information they need about their network to ensure optimal performance and avoid many of the problems discussed in this paper.

Great timing is just the start…

This is when accuracy counts as much as precision: network clients should be synced to the correct time, not just any time, and this is where features that protect against UTC loss, in particular:

- Oscillator upgrade options (to increase holdover precision if GNSS is lost)
- Peering and holdover (also in case GNSS is lost)
- Support multiple GNSS constellations (if primary GNSS is lost)

# Transaction or Distributed Processing

Time synchronization in transaction processing is not new, especially when processing is distributed among cooperating systems on a network. IBM has recognized that time synchronization is critical to execute very high value transactions since the 1960s. According to IBM's Redbook:

> *"There has been a longstanding requirement for accurate time and date information in data processing. As single systems have been replaced by multiple, coupled systems, this need has evolved into a requirement for both accurate and consistent clocks among these systems."*

Today we use many servers and network devices of different types and functions, all networked together, to perform a variety of transactions. Application timestamps tend to have a 1 second lower limit (absolute), and are used to determine when any given transaction occurred (when a purchase order was issued, when the phone call was connected and completed, for instance).

The need for millisecond or microsecond timestamp accuracy derives from the need to execute transactions in a correct sequence, particularly if many transactions occur almost simultaneously. Because computer operations happen automatically and quickly, system clock resolution must be less than the minimum transaction composition and transmission time - resulting in a need for sub-millisecond resolution.

Great timing is just the start…

Keeping widely geographically distributed time servers in tight sync requires very stable oscillators, peering and holdover features, and support for multiple GNSS constellations.

# Software Development

While software development can involve work on different servers and at multiple geographic locations, the code is eventually compiled into a single program. A `make` file function or version control system of some sort is used to manage the compilation of the software from the distributed servers. File timestamps are used to decide what files need to be rebuilt when the underlying source file has been changed. If some of the directories are NFS mounted and the server and client have different notions of the current time, then the `make` function can fail to rebuild some derived objects and so produce an executable that is not based on the most up-to-date sources.

Great timing is just the start…

A reliable UTC time sync is paramount for avoiding these issues, and best provided by multiple GNSS constellation support and OCXO or rubidium upgrade options.

# Email

Email is the de facto standard of written business communication. Every email message that passes across the network bears the originator's timestamp. If that timestamp is incorrect, it can create confusion on the part of the recipient and challenge the credibility of the originating organization. Email timestamps are also critical for establishing the sequence of who said what in a communication exchange.

Great timing is just the start…

Overall reliability features – such as dual power supplies and extended environmental specifications – are crucial for ensuring that email timestamps stay accurate to within at least one second.

# Legal and Regulatory Requirements

Accurate and traceable network time is sometimes a requirement of industry regulations. In the United States, for instance, the Financial Authority Regulatory Authority (FINRA) that governs stock trading requires its members to timestamp stock trades (with an accuracy of one second or better) traceable to UTC at the National Institute of Standards and Technology (NIST). This represents a very large synchronization challenge because many trading firms across the U.S. are subject to this regulation. The reason for synchronized, traceable time in this application is to validate when a transaction occurred for the purposes of order auditing. Other industries like legal, medical, and telecommunications are also expected to adopt traceable time standards as part of their network operating policies.

Great timing is just the start…

Hardware timestamping and a rubidium oscillator upgrade option are especially important for assuring ultra-accurate timestamps in such extremely time-sensitive applications.

# Conclusion: Keeping Time Safe and Effective

The network operations and applications discussed in this paper show that if the need for quality timekeeping is pervasive across the network, then so too is the need to keep timekeeping safe and easy-to-deploy and use. Redundant time sources, holdover schemes, power sources, authentication options, and hardware components very much serve the primary mission of accurate, reliable, and secure time services to the network, as do features that make timekeeping easier to monitor and control (including an intuitive web interface and support for SNMP so the server can alert administrators to out-of-bounds conditions such as a DoS packet flood). While it is certainly possible to source time for free from an Internet time server that lacks these critical attributes, it is important to consider the critical network and business operations that hinge on such a fundamental and essential attribute as accurate, secure, and reliable time.

Remember, great timing is just the start!