Time Synchronization Prototype, Server Upgrade Procedure Support & Remote Software Development

Shania R. Sanders[1]
National Aeronautics and Space Administration, John F. Kennedy Space Center, FL, 32899

Networks are roadways of communication that connect devices. Like all roadways, there are rules and regulations that govern whatever (information in this case) travels along them. One type of rule that is commonly used is called a protocol. More specifically, a protocol is a standard that specifies how data should be transmitted over a network. The project outlined in this document seeks to implement one protocol in particular, Precision Time Protocol, within the Kennedy Ground Control Subsystem network at Kennedy Space Center. This document also summarizes work completed for server upgrades, remote software developer training and how all three assignments demonstrated the importance of accountability and security.

Abbreviations

KSC	=	Kennedy Space Center
KGCS	=	Kennedy Ground Control Subsystem
LCC	=	Launch Control Center
OS	=	Operating System
PTP	=	Precision Time Protocol
NTP	=	Network Time Protocol
TCP	=	Transmission Control Protocol
IP	=	Internet Protocol
UDP	=	User Datagram Protocol
DHCP	=	Dynamic Host Configuration Protocol
PLC	=	Programmable Logic Controller
IEEE	=	Institute of Electrical and Electronics Engineers
CLI	=	Command Line Interface
VLAN	=	Virtual Local Area Network
ms	=	milliseconds
ns	=	nanoseconds

I. Introduction

The goals of this internship were to support the Kennedy Ground Control Subsystem (KGCS) and to obtain training in remote software development. The high level goals of completing tasks included obtaining a better understanding of accountability and security.

Accountability is seen as the need to keep track of the events occurring on devices within the subsystem. This helps improve quality control and system management via backtracking. In other words, if something goes wrong, there is a precise log for each device. The user can utilize this log to find when and where the undesirable event occurred. Truly effective time logs require the clocks of the individual devices to be in sync within a certain accuracy. For devices on a network, there are protocols that can be used to achieve this goal. Currently a protocol called Network Time Protocol (NTP) is used to help sync the clocks of devices on a network. It has a peak time transfer error of >1ms and is inexpensive (compared to Precision Time Protocol, PTP, which will be described momentarily). Unfortunately, the number of viable methods available to gather metrics on the protocol are limited. This will not be useful in gathering data for the proof of concept.

On the other hand, PTP, although more expensive, is a time protocol meant for precision synchronization that is faster (peak time transfer error >100 ns) and has more "extensive…band metrics for monitoring and management"

[1] Remote Software Developer/KGCS Intern, NE-C1/E7, Kennedy Space Center, Colorado Technical University.

[5]. KGCS seeks to implement PTP on devices within the subsystem in order to improve data logs and timing precision. To do this, a prototype was designed consisting of a firewall, a boundary clock, a switch and three testing devices. The three testing devices were a programmable logic controller (PLC), a console and a server.

Security is seen as the measures taken to protect a system against internal and external risks. The purpose of the server upgrade procedure was straightforward – update and correct the document for review – but it demonstrated the necessity of upgrading software to maintain internal security. Remote software training also demonstrated the need for security in software design and execution.

II. Timing Synchronization Project

A. Prototype Configuration Overview

The timing signal (a signal that carries the standard time for everything on KSC networks) comes from what is known as the Grandmaster Clock. This signal comes into a designated area within the subsystem giving KGCS the ability to use it. Being that there might be other unnecessary or hazardous information on the signal, it must pass through a firewall before being accepted onto the subsystem network.

Within the firewall, there are configuration rules that establish what information on the signal should and should not be allowed to pass through from the untrusted network (that is coming from the Grandmaster Clock) to the trusted network (the KGCS network). The rules work on a match/no-match basis; if something on the signal does not match what the firewall says should be there, it is not allowed through. If it does match, then it is allowed through to the trusted side. Once the information passes through the firewall, what remains of the signal (should only be PTP) is passed to a boundary clock.

A boundary clock "is an IEEE 1588 component that allows the synchronization of IEEE 1588 clocks across subnets defined by a router or other devices that blocks the transmission of all IEEE 1588 messages" [2]. The boundary clock has two purposes in this particular configuration. The first is primarily for testing purposes. It is to provide an area to set up a port mirror and make sure nothing besides PTP comes through the firewall. The second is to act as the master clock for the other devices on the system once the configuration is complete. Once the signal has passed through the boundary clock, it is fed into a Brocade switch that distributes the signal to the three tested devices.

B. Grandmaster Clock

As stated previously, the Grandmaster Clock is the source of the timing signal. In order to use this signal, the IP address or the network must be known so that way a port on the firewall can be configured in the same subnet (this is described in more detail later on). When the prototype configuration process first began, there was a period of waiting time between the start date and the date that the IP address could be acquired. To continue working without the real Grandmaster Clock, a simulated one was created using a PLC.

Two Ethernet modules (one to make the connection between the KGCS network and the other to connect to the firewall) were added to a PLC chassis. The module connected to the firewall was given a separate IP address from that of the lab network in order to simulate an untrusted network. In order to configure the Ethernet Modules, RSLinx and RSLogix were used to change the IP addresses and subnet masks.

The PLC was used in this manner until the firewall configuration was almost complete (then the actual signal from the Grandmaster Clock was used). Using the PLC in this way revealed that it is not possible to ping, or "talk to," other systems from the Jun OS Command Line Interface (CLI). This will be described in more detail in the Juniper SRX240 section.

C. Juniper SRX240

The Juniper SRX240 is a service gateway that can be used as a switch or a router. In this configuration, it is acting as a router/firewall. Configuration of the SRX began with factory reset. This restored all defaults including but not limited to Ethernet-switching and Dynamic Host Configuration Protocol (DHCP) service configurations.

Ethernet-switching "eliminate[s] the need for Layer 2 switches in small branch offices and act as an aggregate switch in medium-sized branch offices" [6]. In other words, Ethernet-switching gives the SRX the ability to act as a switch. When this functionality is turned off, each one of the sixteen ports/interfaces on the SRX is isolated from the other ports. Being that this device cannot be configured to act as a router and a switch simultaneously, Ethernet-switching had to be turned off. From the CLI, it was disabled by deleting the functions under each interface description that configured Ethernet-switching.

Note: The SRX operating system (Jun OS) considers Ethernet-switching as a "family." It will be necessary to configure static IP addresses for this prototype configuration (explained in further detail later), and Jun OS will not

allow the user to configure IP addresses when this "family" is enabled. If the user tries, an error will be displayed with the configuration mode of the CLI when the commit is submitted.

DHCP is a protocol which dynamically allocates and assigns network components (i.e. IP addresses) to devices such as consoles, PLCs and servers on the same network. The configuration components are acquired from a DHCP server. In the default factory configuration of the SRX, the SRX ports are assigned IP addresses from a DCHP pool of IP addresses. This presents two issues in the scope of this configuration. First, there should only be two subnets of IP addresses associated with the firewall – that of the untrusted network and that of the trusted network. The default IP Addresses from the DHCP IP Address pool was not a subnet of either of those networks. The second problem is that the ports need to have static IP addresses. The DHCP systems configuration was removed so that each port could be assigned a separate, static IP address.

After completing the steps above and setting the username, password, domain name and hostname for the router/firewall, the next step was to configure each of the interfaces on the device. The SRX has sixteen ports; each received an IP address (except three of the untrusted ports) that was representative of whether it was on the trusted or untrusted network. It is important to note here that whether the IP address is "trusted" or "untrusted" it is only known by the user. The firewall has a set of rules, separate from IP address configuration rules, that define whether a port if trusted or untrusted. Once each port had an IP address, a set of rules were written declaring whether the port is trusted or untrusted.

The next step was to create a bridge, so to speak, between the untrusted and trusted ports so that the signal could be routed into the trusted network. These rules are not to be confused with filtering rules which specify from which IP address the source signal should be coming from and what protocols should be accepted from that signal.

Note: The SRX is not capable of understanding PTP (it is not "PTP smart"). This simply means that the SRX is not programmed to give PTP Ethernet packets priority over other packet types.

Once the configuration was complete, the firewall untrusted port (ge-0/0/0) could be connected to the source of the timing signal. To ensure that the firewall is receiving the signal, an initial thought was to ping the IP address of the untrusted network. If it was received, then that meant that the connection was established. However, this was not possible. The reason why is not clear, however, it can be looked into at a later date. The alternative solution was to monitor the traffic coming into the firewall. Within the CLI operational mode, the traffic coming through the untrusted port (including the untrusted port) could be seen.

Once the configuration was complete, it needed to be tested and there was a need to see exactly what information was coming through on the timing signal. This could be done by configuring a mirror port. Port mirroring is a process where network packets are forwarded from one port/interface to a second interface. The second interface is connected to a sniffer (i.e. Wireshark) that can display the source of packets and the information that they are carrying.

Initially, it was thought to conduct port mirroring on the router/firewall using an unused untrusted port. This was not possible for two reasons. The first reason was that port mirroring requires Ethernet-switching which, as already explained, cannot be used to meet the purposes of this project. The second was that port mirroring with Ethernet-switching is not enabled on this particular model of the Juniper system. So, even if Ethernet-switching could be used in the prototype configuration, port mirroring could not be achieved because this model does not support it. The alternative solution was to set up a mirror port on the boundary clock.

Figure 1. Juniper SRX 240 [3].

D. Allen-Bradley Stratix 8000 Boundary Clock

The boundary clock is crucial to the functionality of the prototype configuration. The first reason was discussed at the end of the JuniperSRX240 section – port mirroring. Due to the fact that port mirroring is not possible on the firewall, a switch (like the Stratix 8000) is needed to connect to the firewall, monitor traffic and determine whether

or not the firewall is configured properly. However, there is still the argument that it could be removed after firewall testing has completed. This is not true.

The Stratix 8000 is needed to act as a master and boundary for the final system which can help to synchronize time on the signal on the trusted side of the network. One of the main goals of this project is to synchronize time among devices in a system. As a master/boundary clock, the Stratix 8000 will act as a standard. For instance, if the times of the devices on the network randomly change, the boundary clock can synchronize the times of other IEEE 1588 clocks to what they should be.

Without the boundary clock, if something were to happen to the timing of end devices, they could synchronize among themselves where one (i.e. the PLC) becomes the master and the other two devices become the slaves (i.e. the consoles and the servers). However, it may be better to have a separate device perform this function for better accountability; if something happens, the user will know for certain who the new master is. It also promotes modularity which, at least for the prototype, will ease troubleshooting efforts. Essentially, the boundary clock is needed for verification during testing and control during implementation.

Figure 2. Allen-Bradley Stratix 8000 [1].

E. Brocade Switch

The Brocade switch is used to distribute a signal to other devices via an Ethernet connection.

F. Allen-Bradley Programmable Logic Controllers, Windows Server and Console/Workstation

To test the timing with the new PTP signal, testing is being conducted on a PLC chassis, a Windows server and a console. In order for the testing to work, all of the devices need to be PTP smart. If they are not able to understand the signal, they will not be able to sync with other devices according to the PTP standards. Although the Allen-Bradley PLCs have the ability to understand what PTP is, the console and the server do not. Therefore, third party software, called Domain Time II, was procured and installed on the server and the console giving them the capability to function according to PTP.

G. Results

At close of business on November 3, 2014, the rules and configurations for the firewall had been established, the Stratix 8000 was set up but not fully configured and it was connected to Wireshark for sniffing. The Domain Time II software was installed and configured on the server and console and documentation was written detailing configuration and troubleshooting procedures.

III. Remote Software Developer Training

As a remote software developer, the first three weeks of the internship included training for this topic. This included learning how to use the Linux virtual machines to access software needed to write ACL scripts as well as design and test remote displays for the firing room. Later in the term, remote developer training was completed in the Launch Control Center which outlined how to interact with software in the firing room (and how it interacted with the local side). This training demonstrated how to read data from displays in real time and how to take metrics.

It also demonstrated the need for internal security to preserve the integrity of a procedure. One such example includes the lock out functionality between remote and local systems. When one side is working on a system, the other is locked out until the other side releases functionality. This prevents both sides from controlling the systems at

the same time which could cause work to be destroyed or overridden (loss of integrity). Having either the local or the remote side log in one at a time also promotes better accountability. For example, if both could log in and work on a subsystem at the same time, depending on the times processes were executed, it could be difficult to track who does what and when. If something went wrong, then it would be even more challenging to backtrack and see where the problem occurred.

IV. Launch Control Center Server Upgrades

The first six weeks in KGCS included working on the time synchronization project; however, it also involved becoming familiar with servers in the LCC that are scheduled for upgrades. Familiarization involved studying CAD drawings and reading through the initial written draft of the upgrade documentation. This knowledge was then used to walk through the server procedures and check the logic of the procedure, grammar and the format. The first set of documentation was completed and submitted for internal review. Once internal review for content and format has been completed, it will be submitted for an external review before testing can begin. Working with the server upgrade procedures illustrated the how to implement security measures and how to balance authentication and confidentiality requirements without having to use specialized software.

V. Conclusion

In the future the timing prototype will need to be completed and the functionality of the configured rules needs to be confirmed. Once that is completed, the primary testing for whether or not the system can sync using PTP can begin. There is also work to be done on the server upgrade documentation. Once the initial internal review is completed, it will have to be reviewed externally. Once that process is complete, actual testing can begin. Finally, for remote software development, as there are no other scheduled tasks or trainings to complete, this portion of the internship has been completed.

Acknowledgments

Shania R. Sanders would like to thank her mentors Everett R. Martin, Kurt W. Leucht, Elias Victor, Tuan A. Le, the Kennedy Space Center Intern Coordinators and the Universities Space Research Association (USRA).

References

1 “AB Stratix Ethernet Switches,” Nelson Electric Supply Company, Inc. [website], URL: http://webstore.nelson-electric.com/ProductCategory.asp?ProductMenuSys=101606 [cited 4 November 2014].

2 “Boundary Clocks,” NIST Engineering Laboratory [website], URL: www.nist.gov/el/isd/ieee/boundaryclocks1588.cfm [cited 4 November 2014].

3 “Juniper Networks SRX240H2 Firewall/VPN, 2GB DRAM,” 1st Advance Solve the Complex. [website], URL: http://www.1stadvance.com/images/srx240.png [cited 4 November 2014].

4 “Juniper Networks TechLibrary,” Juniper Networks [online database], URL: http://www.juniper.net/techpubs/ [cited 4 November 2014].

5 “NTP and PTP (IEEE 1588) A Brief Comparison,” Symmetric.com [online presentation], URL: http://www.en4tel.com/pdfs/NTPandPTP-A-Brief-Comparison.pdf [cited 4 November 2014].

6 “SRX Getting Started - Configure Ethernet ports for switching,” Juniper Networks [website], URL: http://kb.juniper.net/InfoCenter/index?page=content&id=KB16667 [cited 4 November 2014].

7 “Stratix 8000 and 8300 Ethernet Managed Switches,” Allen-Bradley [electronic publication], URL: http://literature.rockwellautomation.com/idc/groups/literature/documents/um/1783-um003_-en-p.pdf [cited 4 November 2014].