Observing Industrial Control System Attacks Launched Via Metasploit Framework

Nathan Wallace
Louisiana Tech University
Ruston, Louisiana USA
nsw004@latech.edu

Travis Atkison
Louisiana Tech University
Ruston, Louisiana USA
atkison@latech.edu

ABSTRACT

Industrial Control Systems (ICS) are present across many industries ranging from automotive to utilities. These systems have been found to be connected to corporate enterprise servers and can communicate over unencrypted communication channels. Interconnections of this type provide an attack vector for people with malicious intent and therefore are a critical cyber security risk. To better understand these risks and possible security measures, this research presents as proof of concept several attacks against a programmable logic controller along with observations made during the attacks. Our results indicate a time sequence difference between legitimate and spoofed command and control packets. Attacks are launched using the Metasploit Framework against a simulated control scenario. Using the observations made in this paper it is then suggested that several features can be extracted and then utilized in next generation mitigation and detection techniques for the industrial control environment.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Network**]: Security and Protection; B.8.1 [**Performance and Reliability**]: Reliability, Testing, and Fault-Tolerance

General Terms

Algorithms, Reliability, Security

Keywords

SCADA, Industrial Control Systems, Intrusion Detection

1. INTRODUCTION

Industrial control systems (ICS) can be found across several industries ranging from transportation to utilities. ICSs are comprised of multiple controllers each operating as logic engines that conditionally cause a mechanism to perform an action. Today's ICSs have seen an influx of solid-state devices with Internet/Intranet networking capabilities. Benefits of this influx include the command and control ability granted to the governing ICS. However, with this influx of smart network capable devices, the potential for various cyber threats arises and therefore sophisticated cyber detection methods must be developed and analyzed. In Miller et al. [1] a survey of SCADA and critical infrastructure incidents was conducted, bringing to light the risks these systems present and a call of need for the mechanisms to secure them.

One of the goals in the current administrations A Policy Framework for the 21st Century Grid is a power infrastructure with improved grid security and resilience [5]. This resilience describes a need for a self- healing network that can prevent, mitigate, and effectively avoid most cyber threats found in a control system environment. A system that can recognize cyber events including malicious intrusions and mitigate them at the control level will effectively prevent most attacks from occurring or at the very least minimize the damage caused by the attack, consequently adapting and training the system to avoid the threat.

At BLACK HAT USA 2011 Las Vegas, security researcher Dillon Beresford presented and demonstrated several attacks against the Simatic S7-1200 programable logic controller (PLC). This paper demonstrates some of these attacks as proof of concept attacks performed against a programmable logic controller (PLC). We then go on to describe observations made during the attacks, focusing on the time differences between legitimate and spoofed command and control packets. Lastly we explore the possibility that this information could be used as part of an intrusion detection system.

2. BACKGROUND

Originally the control system local area network (LAN) was not connected to any Internet/Intranet connected devices. This physically created the so called air-gap, physically securing the devices from other Internet connected devices. However, for an increase in efficiency and remote monitoring capabilities the control system had to be integrated into corporate LANs. Figure 1 shows the Internet connected control system. This controls system protection scheme combines the practices of IT security to the control system with the goal of an end-all secure environment. Furthermore, in a recent report [2], a total of 7,200 control devices have been found to be directly connected to the World Wide Web. This project known as SHINE was stared in Spring of 2012 and sought to raise awareness for controls system security with initial finding reaching 500,000 Internet connected devices.

With that, the traditional IT securing methodologies are constantly being breached with zero day attacks and generally can go undetected on a network depending on the severity and nature of the attack. Therefore if this marriage of traditional IT with control

ACMSE'13, April 4-6, 2013, Savannah, GA, USA.

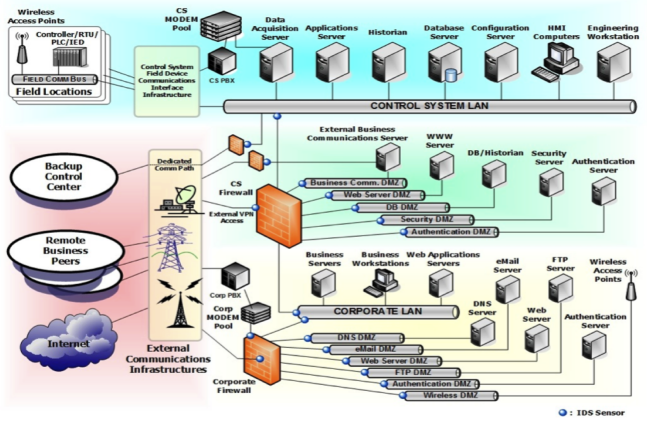


Figure 1: Source: CSSP Current protection of corporate and control domains. [6]

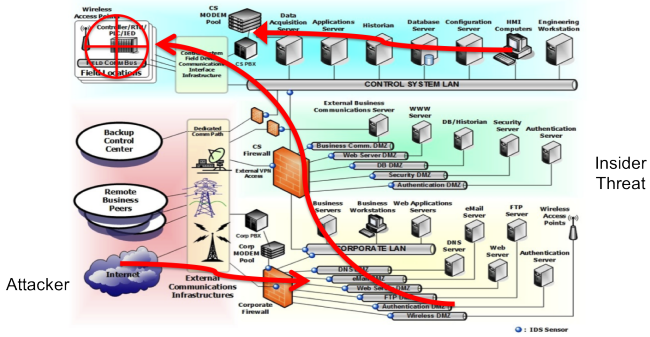


Figure 2: Experimental Assumption: Insider is in the control LAN. [6]

systems is to take place there has to be safe guards in place that can detect, prevent, and mitigate an attack implemented in the control LAN. This research hopes to offer the foundation for anomaly based intrusion detection techniques.

3. EXPERIMENTAL SETUP

The experimental setup for testing involves four machines: the engineer's work station, the attackers work station, the monitoring workstation, and the Simatic S7-1200 PLC. The engineer's work station comes with Siemens Totally Integrated Automation Portal on a Windows 7 machine. The attacking machine is running Backtrack5r3 which has Wireshark, tcpreplay, and the Metasploit Framework 4.5.0 pre-installed. The experimental network topology is shown in Figure 3 and is based on the mirroring functionality of the switch connected the control center to the WAN. All traffic sent to the WAN is then mirrored to the monitoring workstation via the mirrored port. Furthermore, this testing environment assumes that the attacker has already exploited and breached the network firewalls and has gained access to the internal network as shown in Figure 2.

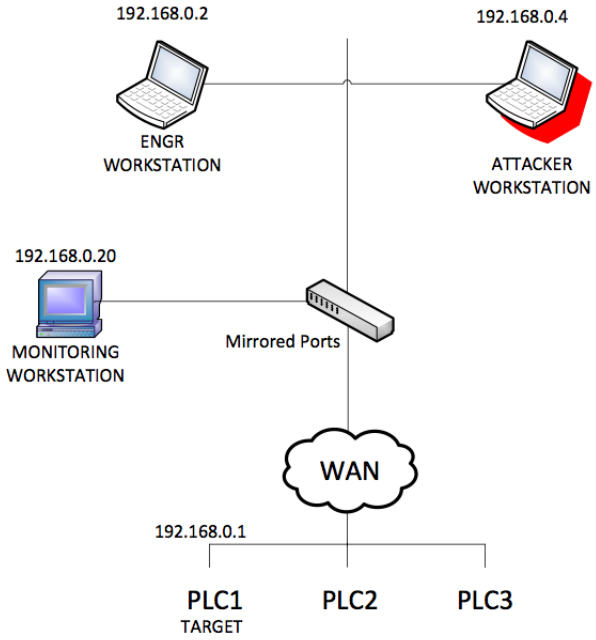


Figure 3: Network Topology Changing Photo

This type of assumption allows for the possibility that the attack can originate from either an outsider or insider threat, someone already inside the corporate network, i.e. disgruntle employee. The devices that encompass a control LAN, have been discovered to be 'accidentally' connected to the internet and are not isolated on an internal network [2]. Therefore, great care should be taken in the networking of these devices. The target is the control devices that actually does the switching or conducts the physical movement, as illustrated by the arrows in Figure 2. Figure 3 shows the network topology for the Testbed with the associated IP address of each control LAN device.

The attack scenario shown in Figure 4 is the test application for our developed attack. This is a critical power application where the load has to have power 24/7 and it is the job of the controller to ensure this task. The controller is programmed to recognize when the utility is no longer supplying the power and will switch to the generator in the event it has lost power. PLCs use ladder logic as the primary programming interface used to set the conditional statements for the controller. Figure 5 shows the ladder logic programmed by the Engineer and has the following tags: PSrc, GSrc, PSind, Gsind, PHindYes, and PHindNo.

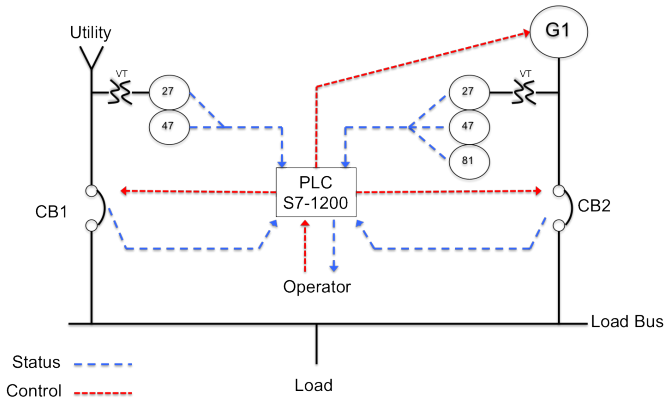


Figure 4: Critical Power Application

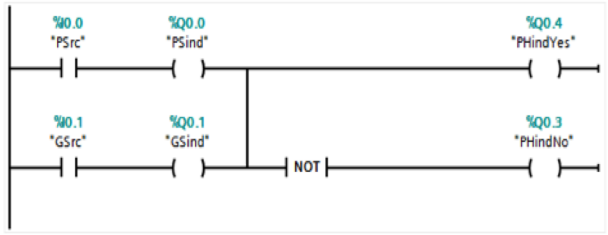


Figure 5: The ladder logic uploaded to the Siemens S7-1200 PLC

4. ATTACKS

Network traffic must first be monitored and captured while the engineering station is communicating with the device. The communication between the engineer and the controller is done through the Siemens' Totally Integrated Automation Portal. While this communication is underway the attacker then monitors the traffic using a network capturing tool i.e. Wireshark. The captured communication will allow the attacker to acquire the handshake needed for device authentication. From the captured tcp streams, packets were loaded into an Metasploit module and then launched against the target PLC. If an attacker has an extensive library of TCP streams, every action that the engineer conducts through the Siemens TIA portal plus more can be conducted by the attacker via a terminal.

The first attack discussed is the initial PLC scan. This scan reveals the make, model, and firmware version of the PLC in question. Such an attack can be used for foot-printing and consequently will enable the following attacks. Figure 6 shows the return string from the crafted Metasploit module. The return string shows the make and model of the PLC (6ES7 212-1BD30-0XB0) and the firmware version (V2.0.).

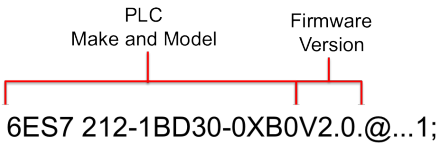


Figure 6: Attack Return String

The next proof of concept attack involves the extraction of the ladder logic tags used to program the controller. Similar to the previous attack this attack involved a series of launched packets targeted at the controller and the return string was the tags listed in 5: PSrc, GSrc, PSind, Gsind, PHindYes, and PHindNo. With this information attackers can rewrite the packets and upload a new program block to the PLC. For instance if an attacker wanted to instead have the output to the GSind instead of PSind, which is implemented in the current control application Figure 4, the attacker can exchange the tags tag in the attack payload. This can be devastating in the instance these outputs (lights) represent physical action or control system alarms that trigger other events.

The last attack involves a toggling of the CPU, basically turning off or on the main program block. The control engineer has the ability to turn off and the main program block of the controller. This functionality granted to the control engineer is assumed to be for troubleshooting capabilities. However, this feature if exploited can be used to issue denial of service (dos) attacks against a control environment. By extracting the packet payloads and placing them in a Metasploit module the same functionality can be granted to the attacker allowing them to either turn off or on the controller's main program block.

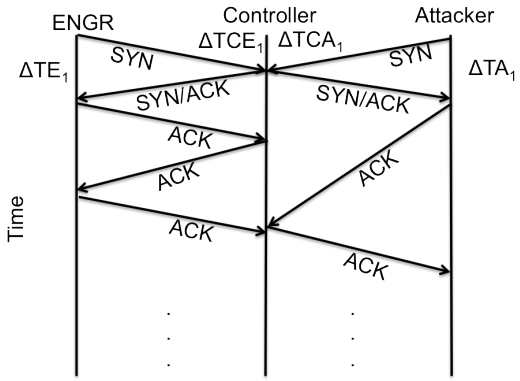


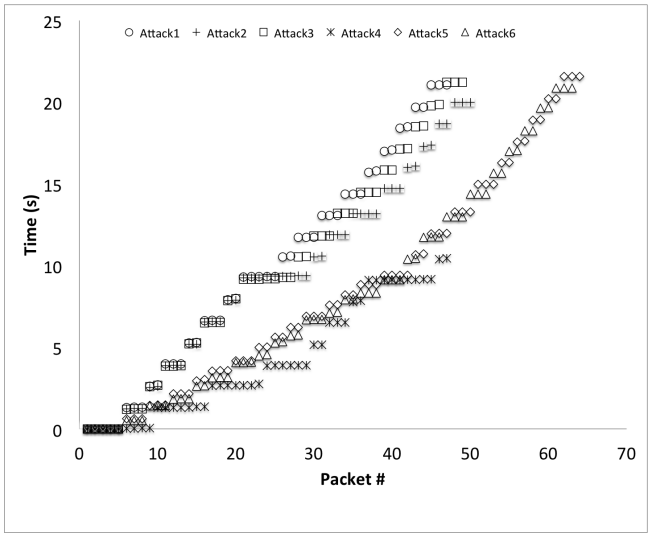
Figure 7: Comparison of ENGR and Attacker Packets

5. FINDINGS

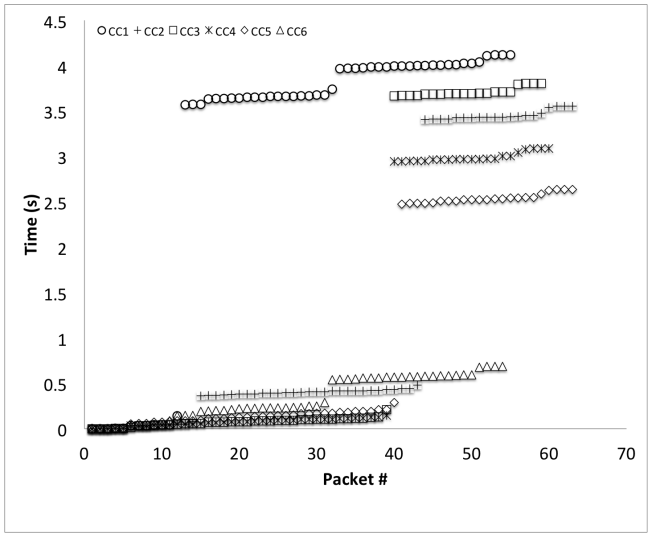
While the attacks described in the previous sections were being launched the network traffic was being captured using the network analysis tool Wireshark by the monitoring workstation as shown in Figure 3. This allowed for a direct comparison in the time sequencing information between legitimate and spoofed packets. Due to the severity of the attack and the corresponding functionality granted to the Engineer the dos attack was used as the basis for comparison between legitimate and spoofed packets from the network traffic.

A custom python script was used to extract the conversation streams based on the TCP/IP SYN flag of the start packet and the ACK flag of the subsequent packets in that conversation. The script also provided the difference in the timestamp of start SYN packets and all ACK packets in the TCP stream. This concept is explicitly shown in Figure 7 with timestamps being noted as ΔT with the suffix ΔE_1 denoting the time the first ACK packet was sent from the engineer (E). Similarly, ΔTCA_1 denotes the timestamp of the first packet sent from the controller (C) to the attacker (A). All timestamps are referenced from the start of the TCP/IP stream, when either the engineer or the attacker begins communicating with the controller.

In comparing the legitimate and spoofed packet streams substantial time differences were found and the resulting TCP streams from both the attacker and engineer are plotted in Figures 8a and 8b respectively. A total of six streams were plotted for both the attack and the command and control (CC) conversations to the controller. This observation can be numerical shown, where the timestamp for two streams are compared side by side. The legitimate stream Figure 8a is a plot of the timestamp information of the packets from the control engineer while the attack timestamps, Figure 8b is a plot of the timestamps from the attacker stream. The delay shown is a direct result of the nature the attack is being conducted. The attack was done using Metasploit modules and as a result operate within the confines of the software application. Each packet has to be individually crafted with the payload stripped from the original control engineer's TCP packet stream. These findings suggest a unit of measure that may be used in future detection schemes which can differentiate between legitimate and spoofed command and control packets.



(a) Attack TCP Streams



(b) Engineer TCP Streams

Figure 8: Time Sequence for Multiple TCP Streams

6. FUTURE WORK

The previous section shows the observed differences between legitimate and spoofed packets. The future work includes the possible classification of Metasploit launched attacks based on time sequencing information and a detailed mathematical model used to define the legitimate communication streams from the control engineer to the controller. This model will be based on advanced feature definitions as defined by the time sequence information and can be implemented in an intrusion detections system (IDS). An IDS similar to the modbus implementation demonstrated by Morris et.al [3] could be implemented using the findings discussed in this paper using the SNORT IDS [4]. Using the model, any marginal deviation off the legitimate feature definitions will trigger an alarm as it is statistically unlikely that that stream would be from the control engineer. An IDS with this functionality is possible given the limited number of machines that will directly communicate with the controller.

7. CONCLUSION

In this paper we develop a simulated control scenario wherein we perform multiple proof of concept attacks against a programmable logic controller. Using a denial of service attack an observation is made comparing legitimate and spoofed command and control packets directed towards a programmable logic controller. Attacks are crafted and launched via the open source framework Metasploit. By analyzing the time sequence information between legitimate and spoofed command and control packets, data suggests a substantial time difference. Furthermore, it is suggested that by defining a set of features based on the observations found in this paper a sophisticated intrusion detection system can be designed for the industrial control system environment.

8. ACKNOWLEDGEMENTS

This research was supported by a Louisiana Board of Regents Graduate Fellowship.

9. REFERENCES

[1] Miller Bill and Rowe Dale. A survey scada of and critical infrastructure incidents, 2012.

[2] ICS-CERT. Project shine. *ICS-CERT Newsletter Monthly Monitor*, October-December, 2012.

[3] T. Morris, R. Vaughn, and Y. Dandass. A retrofit network intrusion detection system for modbus rtu and ascii industrial control systems. In *System Science (HICSS), 2012 45th Hawaii International Conference on*, pages 2338 –2345, jan. 2012.

[4] Rafeeq Ur Rehamn. *Intrusion Detection Systems with Snort, Advanced Intrusion Techniques using Snort, PHP, MySQL, Apache and ACID*. Pearson Education, 2003.

[5] National Science and Technology Council. A policy framework for the 21st century grid: Enabling our secure energy future. June 2011.

[6] Dept. Homeland Security. Recommended practice: Improving industrial control systems cybersecurity with defense-in-depth strategies, 2009.