Exploratory Analysis of Modbus and general IT network flows in Water SCADA System

Anvit Srivastav
Department of Computer Science
Univeristy of Texas at Dallas
Richardson, TX 75080

Carlos Ortega
Department of Computer Science
University of Illinois, Chicago
Chicago, IL 60607

Priya Ahuja
Department of Computer Science
Univeristy of Texas at Dallas
Richardson, TX 75080

Michael Christian
Department of Computer Science
Southern Wesleyan University
Central, SC 29630

Alvaro A. Cardenas
Department of Computer Science
Univeristy of Texas at Dallas
Richardson, TX 75080

ABSTRACT

Monitoring computer network communications is an essential component for detecting suspicious activities in industrial control networks.

There is a growing literature on intrusion detection leveraging the collection of network traces for industrial control systems. Most of these efforts rely on building a profile of normal network activity and then use it to flag anomalies and outliers as potential events worth of investigation.

In this paper we analyze the SCADA network traffic from a water treatment system in Texas and show differences between regular IT network packets, and SCADA-specific network packets. These unique properties of SCADA network traffic can be used to build profiles of normal behavior and then used to identify future anomalies not conforming with expected patterns.

1. INTRODUCTION

Monitoring the activity of computer networks is an essential step for detecting computer attacks (intrusion detection). Intrusion detection systems can work on a "blacklist" approach (deny known-bad connections) or on a "whitelist" approach (only allow known-good connections).

Most of the network intrusion detection tools available for general Information Technology (IT) systems rely on *signatures* (a detection rule that describes how a specific attack can be detected; i.e., a known-bad) of known attacks: once a new attack is discovered, information sharing and analysis centers distribute signatures of the attack so that companies can configure their intrusion detection/prevention systems to identify these attacks and block or terminate these malicious connections. This methodology works well for attacks that are widely disseminated; however, it does not work well for unknown or targeted attacks. Because the technology underlying industrial control systems is highly heterogeneous and because attacks that want to achieve a physical consequence in control systems (e.g., causing a spill, or equipment damage) need to be tailored specifically for the intended target, we anticipate that most attacks against industrial control systems will be unique to the intended target. Therefore intrusion detection in industrial control systems cannot rely on the dissemination of signatures of attacks (i.e., it cannot rely on known-bad).

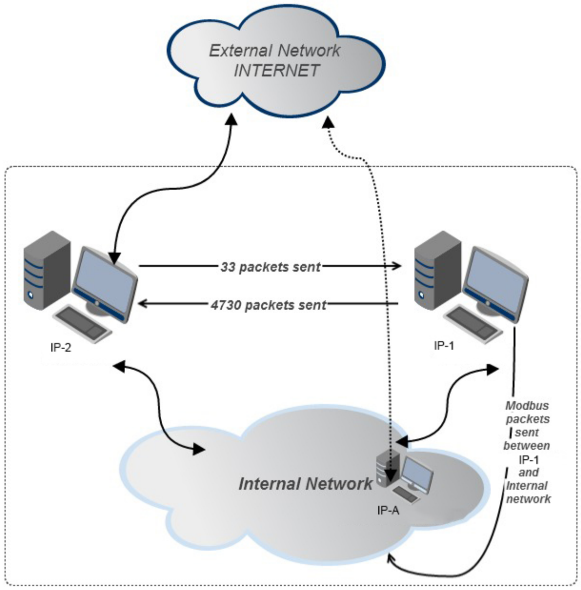

Figure 1: Network Architecture

An alternative to using signatures of known attacks is to create a whitelist of known-good connections and only allow those specific connections. Creating a whitelist manually is error-prone and time-consuming, so in this paper we ex-

ICSS '15, December 06, 2015, Los Angeles, CA, USA
© 2015 ACM. ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

plore how accurate would whitelists be if they are generated from building a profile of network traffic from normal flows from data obtained from the computer networks of a water treatment plant in Texas.

2. RELATED WORK

One of the first papers to consider intrusion detection in industrial control networks was Cheung et al. [3]. Their work articulated that network anomaly detection might be more effective in control networks where communication patterns are more regular and stable than in traditional IT networks. Similar work has been done in control systems [4, 6], smart grid networks [1, 5] and in general CPS systems [7]

The analysis done in this paper is motivated by the recent Ph.D. dissertation of R. Barbosa on traffic analysis in industrial control systems [2]. In particular we apply the same network traffic analysis tools he explored to another dataset of industrial control systems. Our analysis also assumes that the network traces we obtained from the Texas water treatment plant do not have any malicious packets in the Modbus traffic.

3. NETWORK ARCHITECTURE

Figure 1 summarizes the network architecture that was obtained after studying and getting results from a 24-hour packet capture trace in an industrial control network.

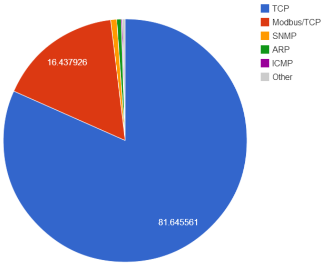

Figure 2: Percentage of different protocol types in the network trace we analyze.

There were different types of network protocols in our packet capture like TCP, UDP, ICMP, IGMP, SNMP and one particular industrial control network protocol: Modbus over TCP. Figure 2 shows the distribution of different protocols we observed; and Table

In this paper we refer to computers by their anonymized Internet Protocol (IP) number. As shown in Figure 1, there are 2 computers that are involved in most of the communications in the network: IP-1 and IP-2. Both, IP-1 and IP-2 talk each other as well as other Internal network IPs.

All communication with the outside Internet was done only through IP-2 and in smaller part through another IP address we will call IP-A (this might be a Proxy). In addition, IP-2 and IP-A were not at all involved in any Modbus communications (a good sign).

Protocol	Percentage
TCP	81.645561
Modbus/TCP	16.437926
SNMP	0.798312
ARP	0.578608
ICMP	0.151973
0x8874	0.122386
NBNS	0.052091
UDP	0.031368
HSRP	0.030064
SMB	0.029413
SAMR	0.024677
DN	0.013685
DHCPv6	0.010818
SMB2	0.007777
DCERPC	0.00517
CLDAP	0.00391
BROWSER	0.003041
LDAP	0.002998
NTP	0.001651

Table 1: Percentage of each network protocol in our network sample.

We are most interested in how the Modbus traffic compares to the general IT network traffic as we would like to create whitelist profiles for Modbus data.

Figure 4 shows Modbus data flows between IP-1 and the internal network: IP-1 is the central node, and as we can see, it communicated with almost all nodes in the internal network. The nodes are color coded based on indegree where white is 0, light blue is 1, medium blue is 2, and dark blue is 3. These represent the number of sources that connect to that particular node. There are 7 nodes that the central server (IP-1) does not communicate with. It is also possible to see that some RTUs communicate with each other.

The devices with whom IP-1 communicates are Remote Terminal Units (RTUs) but they are represented as computers for generality. Whereas, there are few communication in which IP-1 is not involved. Communications which do not involve IP-1 have been shown in Figure 4.

4. COMPARISON BETWEEN IT AND INDUSTRIAL CONTROL NETWORK TRAFFIC

We would now like to study if general IT network communications are different from industrial control (Modbus in this case) communications. Our hope is that Modus communications will be fairly stable and regular when compared to general IT communications, and therefore, will enable us to create better profiles of “normal” Modbus activity that can be used for a whitelist of industrial control communications.

We define *new connections* as a new network flow between two IP addresses we have never seen before. For every network connection, we check if the pair (IP_a, IP_b) have been seen communicating before. If we have seen it before, we ignore it; otherwise we add it to our statistics.

Figure 5 shows that the majority of connections (more than 50%) in our 24-hour trace are created within the first 1.5 hours. After that the connections grow slowly. We also detect an anomaly: there is a sudden surge in connections

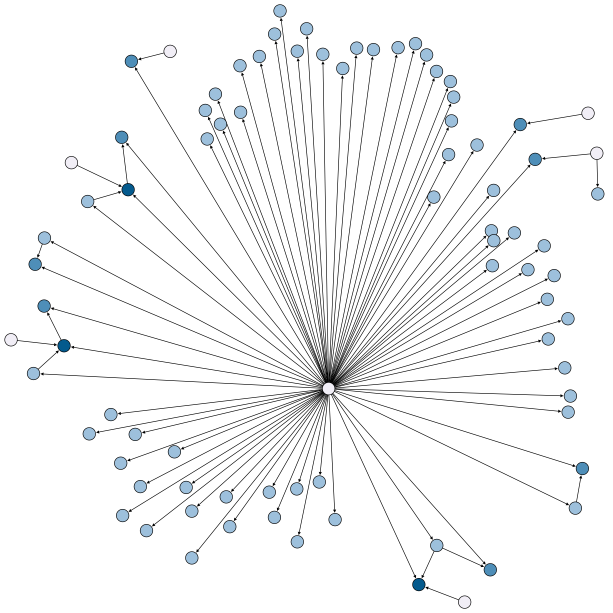

Figure 3: Complete Modbus network

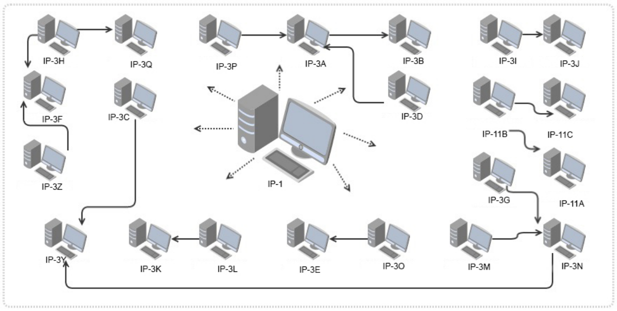

Figure 4: Modbus traffic flows in the internal network.

at 1156 minutes when 111 new connections were formed.

In contrast to general IT connections, if we focus on Modbus communications solely, the number of pairs of IP addresses communicating grows at a much slower pace. As can be seen from Figure 6, almost all IP addresses communicating are discovered in the first 6 seconds (80% of them). After the first 3 mins, there are absolutely no new connections except in the interval 1154-1297 minutes, where 4 small bursts of new connections were observed. It is interesting to note that the first and the second peak occur at 1154 and 1156 minutes, which is almost the same time period where the peak from the non-Modbus connection graph was observed. This results show that while trying to establish a whitelist of allowed connections between two IP addresses might raise many false alarms for general IT traffic, it might be a good solution if we focus on whitelisting Modbus connections between pairs of allowed IP addresses.

A similar analysis can be done by monitoring only the new IP addresses observed in the 24 hours (not the new IP pairs). Figure 7 shows the amount of new IP addresses for general IT connections, and Figure 8 shows the number of new IP addresses exchanging Modbus packets. The fact that these to figures are very similar to Figure 5 and Figure 6 tells us that doing a fine-grain whitelist using allowable IP pairs rather than whitelisting an IP address without taking into account with whom it is communicating with, is a better approach. By whitelisting only IP address pairs we won't get a large number of false alarms and still minimize the opportunities for attackers in creating malicious connections.

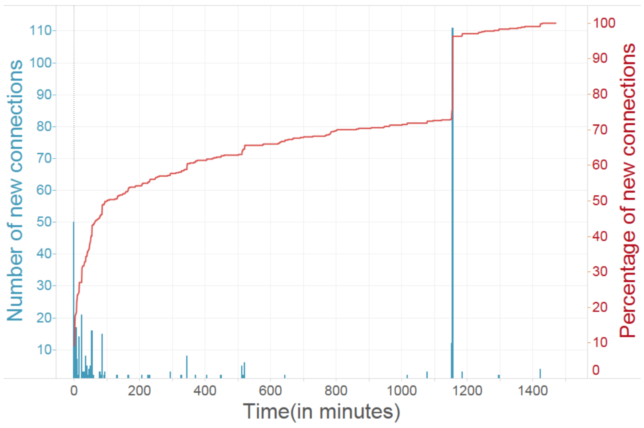

Figure 5: New Non-Modbus Internet Connections per Minute.

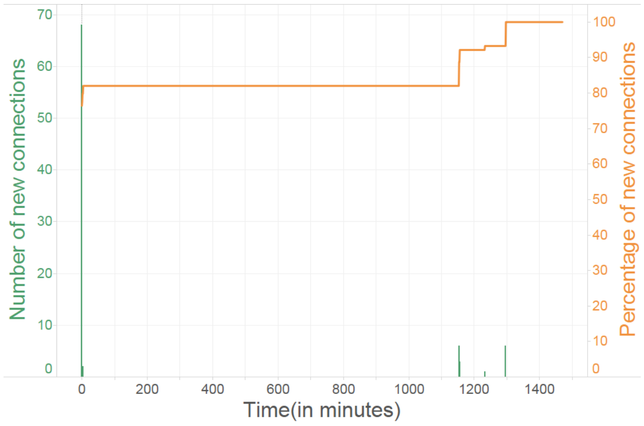

Figure 6: New Modbus Connections per Minute.

In addition to monitoring the IP network connectivity we can also monitor the frequency of communications to determine if they are regular enough to help us identify suspicious activity (e.g., a spike in network traffic).

As seen in Figure 9, the number of bytes being for in general IT connections fluctuates but for the most part, the number of bytes being sent per minute stays under 20,000 and it shoots up close to 30,000-40,000 intermittently. There are few peaks which shot above even 50,000 bytes a minute in the range 1115-1220 minutes.

On the other hand, we can see in Figure 10, the number of bytes transferred over Modbus stayed within 5,000 and

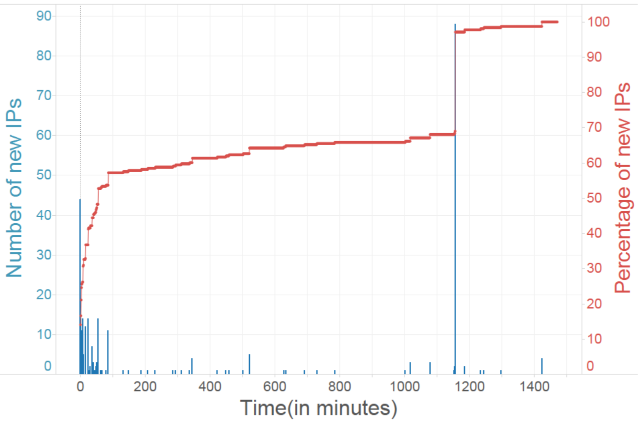

Figure 7: New Non-Modbus IPs per Minute

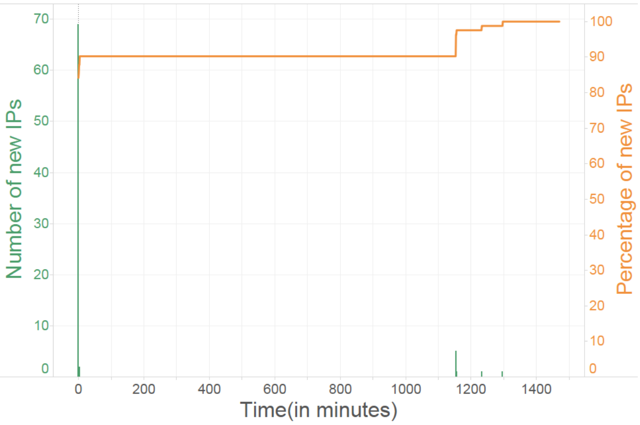

Figure 8: New Modbus IPs per Minute

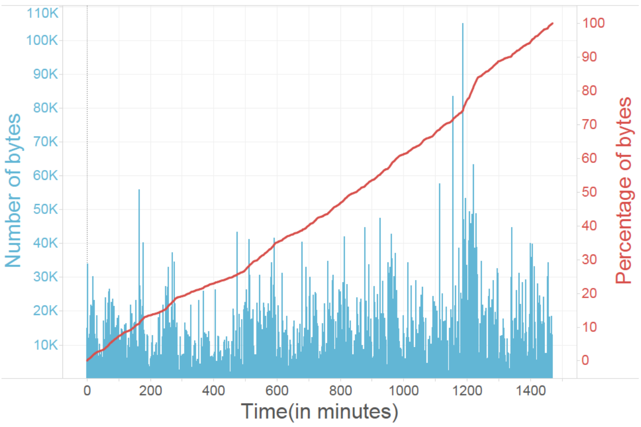

Figure 9: Number of bytes per minute in general IT communications.

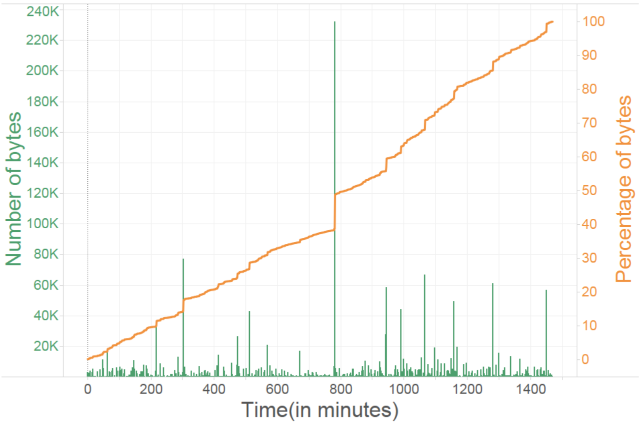

Figure 10: Number of Modbus bytes per minute.

it crossed 20,000 only 13 times. However, There is a huge peak at 782 minutes when 232,314 bytes were transferred. If we had a whitelist allowing only 20,000 bytes-per-minute allowed in the Modbus network we would have denied network connections (or raised alarms) 13 times. While keeping track of the amount of bytes being transferred in the Modbus network might give some indicators of anomalies, we still need further research to characterize the size and frequency of "normal" communications.

Instead of looking at the number of bytes per minute, we can also look at the number of connections per minute, or in general, we can look at the number of connections in a given interval of time. Finding an exact interval of time that will reveal any pattern that can help us can be difficult.

Figure 11 shows the number of distinct Mondbus connections at each second: i.e., for each second we count the number of distinct connections for that second. When graphed it is possible to see that there is a uniformity in the way that the network behaves. What is interesting to see is the blue dots, which represents when the number of distinct connections drops significantly. This happens about every 26 secs starting at second 5.

In order to understand better the frequency of communi-

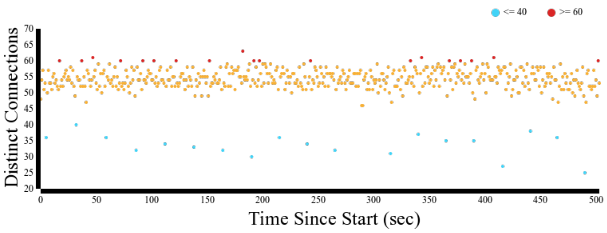

Figure 11: Distinct Modbus Connections at Each Second

cations in Modbus networks we tried to understand the frequency of communications of each individual device—-Remote Terminal Units (RTU). We selected a 1-sec observation period and then counted how long did it take devices to start a Modbus connection. We found the following intervals of communications.

Several RTUs communicated every second. These are potentially critical systems that need constant monitoring.

We also found some RTUs which the server communicates to for the most part every 2 seconds. This further shows how the Modbus network is fairly rigid, controlled, and to an extent predictable.

Another interesting interval that we found was a set of RTUs that communicate with the control server every 300 seconds.

5. CONCLUSIONS

In this paper we have performed a preliminary analysis of the possible ways to create profiles of normal Modbus communications in a control system to use as a whitelist of allowed behavior.

We are currently working to increase the level of monitoring by performing deep-packet inspection of Modbus packets. So in addition to looking at meta-data of packets such as the origin and destination pairs (IP addresses) and the size and frequency of communications, we take a look at the application-layer information in the packets.

Our final goal is to use these network features combined with the sensor data from physical observations (e.g., water levels, pump and valve status, etc.) to provide engineers at the control center of a plant, with information to help them decide on whether or not alarms generated by the physical state of the system are due to random failures or if there is any indication of an attack.

For example, when traditional HMI systems alert operators of a safety problem there is currently no way to identify if this anomaly originated because of a malicious intrusion, or from a random natural fault in the system, and depending of the origin of the safety concern (malicious or accidental), operators will need to react differently.

Operators generally tend to assume that these anomalies are due to natural events, but the growing evidence of cyber-vulnerabilities and attacks to control systems shows that we need to evolve current tools so that operators of physical systems can make an informed decision on how to respond to these alerts.

In particular, our goal is to provide better tools for incident response by enhancing the existing procedure in handling critical situations by providing security contextual information for incident response.

Acknowledgments

The work of Carlos Ortega and Michael Christian was supported by a Distributed Research Experience for Undergraduates (DREU) from the Coalition to Diversity Computing (CDC) and the Computing Research Association (CRA). The work at UT Dallas is supported by the Air Force Office of Scientific Research under award FA-9550-12-1-0077.

6. REFERENCES

[1] Muhammad Qasim Ali and Ehab Al-Shaer. Configuration-based IDS for advanced metering infrastructure. In *Proceedings of the ACM SIGSAC Conference on Computer & Communications Security*, pages 451–462, 2013.

[2] Rafael Ramos Regis Barbosa. *Anomaly detection in SCADA systems: a network based approach.* University of Twente, 2014.

[3] Steven Cheung, Bruno Dutertre, Martin Fong, Ulf Lindqvist, Keith Skinner, and Alfonso Valdes. Using model-based intrusion detection for SCADA networks. In *Proceedings of the SCADA Security Scientific Symposium*, volume 46, pages 1–12, 2007.

[4] Dina Hadžiosmanović, Lorenzo Simionato, Damiano Bolzoni, Emmanuele Zambon, and Sandro Etalle. N-gram against the machine: On the feasibility of the n-gram network analysis for binary protocols. In *Research in Attacks, Intrusions, and Defenses*, pages 354–373. Springer, 2012.

[5] Adam Hahn and Manimaran Govindarasu. Model-based intrustion detection for the smart grid (minds). In *Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop*, page 27. ACM, 2013.

[6] Maryna Krotofil and Dieter Gollmann. Industrial control systems security: What is happening? In *Industrial Informatics (INDIN), 2013 11th IEEE International Conference on*, pages 670–675. IEEE, 2013.

[7] Robert Mitchell and Ing-Ray Chen. A survey of intrusion detection techniques for cyber-physical systems. *ACM Comput. Surv.*, 46(4):55:1–55:29, March 2014.