Smart City Security – Attack and Defense Stratagems

Introduction

* Why your topic is important (convince us!)
* Where is it used? Applications
* What you will talk about / do
* Overview of the rest of your paper

Background and related work

* Any relevant and specific info, e.g. software / hardware statistics, equipment used
* What other people had to say on this topic
* What other people did on this topic
* Problems and shortcomings of their work
* How your work is different and better

Proposed methodology

* Your approach to the problem
* What you did
* Code / Algorithms
* What did / didn’t work
* Results – include graphs, equations, pictures, as appropriate

Conclusions

* What was accomplished / learned
* What you would have done differently
* Future work

References

Appendix

* **Main point:**

Technology Discussion

1. Background on Smart Grid Security
2. False data injection attack
   1. Attack on integrity; Integrity violation
   2. Examples: Puerto Rico, disaster events
   3. Defense includes protection of state estimation or raw meter data
   4. Defense:
3. Popping HMI attack
   1. Target Critical Time Delay <3msec
   2. People can find out information about the network
   3. Defense: Deterministic Browser
4. Privacy Attack
   1. Vehicles/Terminals
   2. Place some sort of privacy algo, present current research on topic
   3. Defense: Hadamard basis to anonymize

Problem Approach

1. Build Deterfox, setup necessary Windows applications to emulate IEC standard and one machine, See if I can find a target machine time
2. Perturb the power grid network dataset either using RAPPOR or other method.

Conclusion

* Write something lel

**Introduction**

Today’s society rely on industrial control systems to provide services like gas, water, and electricity with differing loads of demand and supply. Supervisory Control and Data Acquisition (SCADA) is a common framework used to describe control systems feedback in industrial systems. In the 21st century, SCADA systems adopt increasingly more computers to control industrial processes, and therefore are targets for nation-states, individuals, and organizations.

SCADA systems do not only exist in the industrial plant, but also play a role in building automation systems. Most universities or organizations that manage a collection of buildings within the campus will utilize a networked SCADA system to maintain control over temperature, water, electricity, and mechanical systems such as elevators.

However, all SCADA systems at some point need to be human operable. Whether it is a mechanical switch, a console command, or a web interface, the feedback system needs to have human intervention and control right down to the individual mechanical components. Today’s corporations mainly use the Human Machine Interface (HMI), to represent this component of SCADA and are typically web interfaces that can be accessed through major web browsers.

In this paper, we discuss the development of the HMI, summarize the research that attempts to secure vulnerabilities, and present further protections for the HMI with an implementation created in a SCADA honeypot to model threats.

**Background and related work**

1. *History of SCADA development*

To understand the research and implementations to protect SCADA vulnerabilities, a quick overview of the development of SCADA is required. Industrial control systems developed rapidly as the demand for services like natural gas, water purification, and electricity generation escalated during the 1960s along with technological innovations in both engineering and computers.

SCADA has developed through generations of improvement with new technology. These generations are referred to as the monolithic, distributed, networked, and Internet of Things (IoT) stages [Wagneer, 18]. Monolithic and distributed describe systems that used a mixture of mechanical and command line interfaces. The distributed, networked, and IoT SCADA systems rely on a mixture of present day protocols like HTTP and TCP as well as protocols specified by IEC 60870 like Modbus or DNP3.

Most modern day SCADA systems include Programmable Logic Controllers (PLCs) and Remote Terminal Units (RTUs). These devices control the machines running on the industrial plant based on current and past data readings. Other computers placed either on or off site delegate instructions to PLCs and RTUs in order to meet the supply and demand of the plant. SCADA systems also include data historian and logging servers that serve to track production of the plant and calls to services.

1. *Research and Response*

However, the networks for SCADA systems are very vulnerable to attack since most of the computer to device protocols like BACnet does not include any sort of cryptographic security [Sciencedirect] article] and could not be upgraded to include cryptographic protocols since the data buses are very small. With the invention of the World Wide Web in the 1990s, protocols with more security features could be used to wrap around the vulnerable ones, so most research focused on using cryptography to secure the networks controlling SCADA systems [https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1612765&tag=1]. Controlling the TCP or UDP connections within the network became a priority, and eventually protocols were established like ISA 95/99 and ASME standards and government guides were produced from organizations like NIST, FIPS, and ISO [https://ac.els-cdn.com/S0167404806000514/1-s2.0-S0167404806000514-main.pdf?\_tid=52dce760-cac4-4c08-a1b1-a7500b38aa30&acdnat=1524864535\_1e42ac4b1d32fc50a118a2be11ddda8c]. Current research focuses on models or introduces a new security framework to protect SCADA systems.

1. *Problem of the HMI*

By looking at the SCADA system from a layered view of computers, current and past research has succeeded in modeling, implementing, and improving the protection of networks within SCADA systems. However, the applications that run within these networks are now targeted towards attack. Different systems have different tech stacks, but all need a common interface in which to issue commands either on-site or remotely: the HMI.

In the last two decades, the popularity of web browsers redefines the HMI into a browser-compatible space. All interactions now go through a web application that is easier to modify and update. Companies that create SCADA packages now offer the hardware PLCs and RTUs along with web and mobile accessible pages. While these applications increase usability, web technologies increase the attack space further for SCADA systems more so than that of network protocols, which are slower to change and develop.

Out of the standards and guides researched, only ICS-CERT had any mention of potential vulnerabilities in web applications [https://ics-cert.us-cert.gov/sites/default/files/recommended\_practices/RP\_CaseStudy\_XSS\_20071024\_S508C.pdf] using cross-site scripting. Since there is no research into security vulnerabilities in HMI, there are no codes in any standard or guide and therefore no models in existing tools to detect web application vulnerabilities. If there are any attacks on HMIs in SCADA systems, they are normally reported to ICS-CERT and have dedicated CVEs, such as the case with Siemens WinCC version 7 (CVE-2014-8551).

1. *Attacking the HMI*

In this research, we will implement attacks on the HMI of a mock SCADA setup. The attack will not include compromising authentication or access control and assume the attacker controls an off-site computer with access. The point of the attack is to see if a vulnerable HMI triggers the intrusion detection tools and logs to ensure that a loss of integrity within the system can be tracked.

1. False data injection attack
   1. Attack on integrity; Integrity violation
   2. Examples: Puerto Rico, disaster events
   3. Defense includes protection of state estimation or raw meter data
   4. Defense:
2. Popping HMI attack
   1. Target Critical Time Delay <3msec
   2. People can find out information about the network
   3. Defense: Deterministic Browser
3. Privacy Attack
   1. Vehicles/Terminals
   2. Place some sort of privacy algo, present current research on topic
   3. Defense: Hadamard basis to anonymize

* **Cao, Y., Chen, Z., Li, S. and Wu, S. (2018). *Deterministic Browser*. [online] Arxiv.org. Available at: https://arxiv.org/abs/1708.06774 [Accessed 28 Nov. 2018].**
* **Chen, P., Yang, S., McCann J., et al. (2018). *Detection of false data injection attacks in smart-grid systems - IEEE Journals & Magazine*.**
* **El Mrabet, Z., Kaabouch, N., El Ghazi, H., & El Ghazi, H. (2018). *Cyber-security in smart grid: Survey and challenges.*** Computers & Electrical Engineering*,* 67, 469-482.
* **Liu, Y., Reiter, M. and Ning, P. (2018). *False data injection attacks against state estimation in electric power grids*.**
* **Wang, W. and Zhuo, L. (2018). *Cyber Security in the Smart Grid: Survey and Challenges.*** *Computer Networks: The International Journal of Computer and Telecommunications Networking.* Volume 57 Issue 5, April, 2013. Pages 1344-1371