SUGISAKA MASANORI

Journal of Robotics, Networking and Artificial Life
Vol. 9(3); December (2022), pp. 216–220
ISSN (Online):2352-6386; ISSN (Print):2405-9021
https://alife-robotics.org/jrnal.html

Research Article

Scalable ICS Honeypot Design by Description Files

I-Hsien Liu, Jun-Hao Lin, Hsin-Yu Lai, Jung-Shian Li
Department of Electrical Engineering / Institute of Computer and Communication Engineering, National Cheng Kung University
No.1, University Rd., East Dist., Tainan City 70101, Taiwan

ARTICLE INFO

Article History
Received 28 October 2021
Accepted 07 August 2022

Keywords
Industrial control system
Honeypot
Cybersecurity
Shodan

ABSTRACT

A prototype honeypot system based on the Modbus/TCP protocol is designed for the protection of Industrial Control Systems (ICS). The proposed system operates under the control of a single server and enables multiple agents, each with several honeypot devices, to be deployed in different industrial environments. For each honeypot, the device characteristics are defined by JSON description files. The experimental results show that the interaction behavior of the proposed honeypot is closer to that of an authentic ICS device (a PLC) than that of the Conpot open-source ICS honeypot reported in the literature. Furthermore, the honeypot is awarded a perfect score by the honeypot scoring mechanism of Shodan Internet of Things (IoT) search engine

1. Introduction

With the advancement of communication and network technology, Industrial Control Systems (ICS) [1] have moved beyond traditional small-scale closed operations to Internet-connected environments, causing a gradual blurring of the boundaries between Information Technology (IT) and Operational Technology (OT). However, while this shift in ICS technology brings numerous practical advantages, including better control, higher utilization, and lower costs, it also increases the risk of malicious network intrusion. Cybercrime poses a significant threat to critical infrastructure facilities such as power grids, reservoir and water distribution systems, public transportation systems, and so on. Thus, it is essential to understand the tactics, techniques and procedures (TTPs) adopted by malicious adversaries to develop effective detection and defense mechanisms

Accordingly, the present study proposes a scalable honeypot framework based on the Modbus/TCP protocol [2] for the protection of ICS networks. A set of extensible description files is prepared in the JSON (JavaScript Object Notation) [3] format to mimic the characteristics of typical ICS devices, such as the IP address, protocol, port number, register value, response method, and so on. The resulting "devices" are then deployed as honeypots to attract the attacker's attention, record the communication process with the attacker, gather information about the attack, and generate the information needed to prepare defense strategies.

The feasibility of the proposed honeypot system is demonstrated by comparing the Modbus/TCP interaction behavior of the simulated device with that of a real ICS device and the open-source Conpot honeypot system [4]. The ability of the honeypot device to deceive an attacker and serve as a bait for malicious attack is then evaluated using the Shodan Search Engine [5][6].

Corresponding author's E-mail: ihliu@cans.ee.ncku.edu.tw, jhlin@cans.ee.ncku.edu.tw, hylai@cans.ee.ncku.edu.tw, jsli@mail.ncku.edu.tw
: www.ncku.edu.tw

2. Background

In recent decades, traditional ICS systems, used for local closed-system control and operation purposes, have been increasingly extended to Ethernet-based systems, in which large-scale distributed operations are monitored and controlled remotely through the Internet. However, the operational benefits which such ICS systems bring come at the expense of a greater exposure to malicious attack. For example, in the infamous Stuxnet attack in 2010, multiple centrifuges at a nuclear plant in Iran were destroyed through a multi-part worm spread through MS Windows [7].

Honeypots, in which malicious users are baited to launch an attack such that their attack behaviors can be captured and scrutinized, are regarded as one of the most effective methods for guarding against such attacks [8].

However, with the development of honeypot technology, the attacker's hiding space is oppressed. As a result, they are typically more wary before launching an attack and often make use of an IoT search engine such as Shodan [5][6] to evaluate the probability of the intended target being a honeypot before committing to the attack.

2.1. *ICS Honeypot*

ICS honeypots attempt to imitate ICS devices, such as PLCs, in order to deceive or trap attackers, build various baiting hosts, capture and analyze the attack actions, and understand the attacker's behavior.

Conpot is a low-interactive open-source ICS honeypot developed by The Honeynet Project [4]. It is easy to implement and supports many ICS protocols, including Modbus/TCP, S7comm, EtherNet IP, and so on. However, its fingerprint features are obvious and are easily identified by honeypot detection tools or IoT search engines.

2.2. *IoT Search Engine*

Shodan is an IoT search engine commonly used by attackers to reconnoiter potential attack devices on the Internet. In Shodan, the honeypot probability of an IoT device is rated on a scale of 0-1, where a score of 0 indicates that the device is not a honeypot and a score of 1 indicates that the device is a honeypot [6]. As such, Shodan is a powerful tool in the hands of an attacker and poses a serious threat to the security of ICS networks.

3. Honeypot Design

Existing ICS honeypots such as Conpot offer only limited interactions with malicious attackers. Thus, they are not only easily recognized by Shodan (for example), but also provide only little useful information regarding the attack behavior.

Accordingly, the present study develops an ICS honeypot system which attempts to mimic the more rich reaction response of a real PLC device (e.g., numerical values representing the sensed temperature, humidity, and air pressure conditions). Taking the Modbus/TCP protocol for illustration purposes, a set of description files is prepared using the JSON lightweight data exchange language as the definition of honeypot characteristics. The description file is the core of the honeypot behavior. It is used to imitate the network response behavior of the real PLC. Notably, the description file defines the honeypot IP address, port number, register address and value, and response method, and enables the user to quickly and flexibly configure the honeypot in different industrial control environments.

3.1. *System Architecture*

Figure 1 shows the basic architecture of the proposed honeypot system, consisting of a honeypot controller, several honeypot agents, and multiple honeypots at each agent. As shown, the system operates through a graphical user management interface implemented on the controller, which then transmits honeypot description files to each honeypot agent, controls the opening or closing of individual honeypots at the agents, records the honeypot log information, and so on. During operation, the honeypot agents parse the description files sent from the honeypot controller, generate the corresponding honeypots according to the features listed in the description files, record the visitor information (e.g., the IP address, port number, time, and behavior) in their local databases, and finally take turns in returning this information to the honeypot controller.

3.2. *Description File*

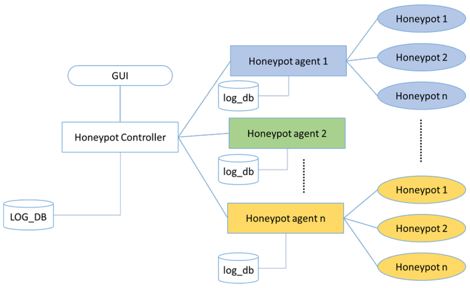

Figure 1 Honeypot System Architecture

The description file is the core of the proposed honeypot system. The use of a description file framework in the present study is motivated by the inherent simplicity of the communication behavior of the equipment in industrial control environments. For example, the communication process often involves nothing more than an HMI reading a sensor value through a PLC, or a sensor sending its signal to the PLC. In other words, under normal circumstances, it is not difficult to imitate the network communication behavior of industrial control equipment. Having conducted a preliminary survey of various real-world industrial control devices, the Modbus/TCP protocol was chosen for illustration purposes. Moreover, the study chose to focus on the surface observed by a PLC in the network. The simulated items included the PLC equipment model, the IP address, the port number, the address and value of each register (i.e., the holding register, input register, input coil, and output coil), common standard function codes (read or write, single or multiple registers), and common exception codes.

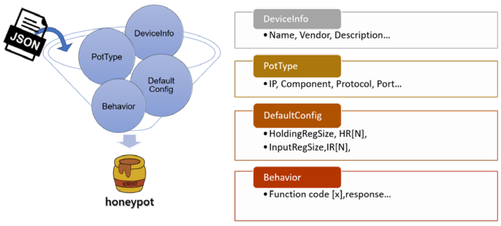

Figure 2 Description File Schema

As shown in Figure 2, the description file was organized into four blocks to simulate the network state of the PLC, namely Device Info, Pot Type, Default Config, and Behavior. Furthermore, extensible reserve space was left in each block for the addition of other special undefined conditions as required.

3.3. *Implementation Sample - Modbus/TCP*

In contrast to other ICS honeypots, the ICS honeypot system proposed in the present study not only provides industrial protocol services, but also focuses on improving the authenticity of the responses. In particular, in accordance with the original Modbus specification [2], a socket structure is used to reproduce the honeypot system that supports the Modbus/TCP protocol.

The content of the description file is parsed by the honeypot agent based on the response mode set by the user through the GUI, and the honeypot service is opened at multiple sites. When a request is subsequently received at one of the honeypots, a check is made to determine whether the request is a Modbus/TCP request. If not, the honeypot simply ignores it and does not respond. Otherwise, the honeypot checks whether the function code is supported and whether the data is normal and if so starts to capture the attack information. The honeypot then replies to the attacker based on the response content specified in the description file. In the event that abnormalities are detected in the function code, data, or data capture process, the honeypot responds with the appropriate exception code. Finally, the entire interaction log is stored in the agent database for further processing by the honeypot controller.

4. Experiments and Results

The experiments commenced by comparing the network responses of a real-world PLC connected to an environmental control model (AMASTek, Taiwan), Conpot [4], and the honeypot system proposed in the present study. The Shodan honeyscore for the proposed honeypot system was then evaluated and compared with that of Conpot.

4.1. *Interactive Behavior Verification*

For the PLC considered in this study, register addresses 0, 1 and 2 hold the sensed values of the temperature humidity, and pressure, respectively.

In the framework proposed in the present study, the response of the honeypot device to the Modbus function code 03 request is defined by the description file. In the present example, the honeypot imitates the response content of the real PLC by filling the ‘Behavior’ part of the description file into the numerical interval and generating random integers at the corresponding register addresses. As shown in Figure 3, it reads the value of the temperature register and records it every minute. Monitor its ambient temperature changes, turn on the air conditioner after 50 minutes and set the temperature to 25 degrees Celsius. It can be seen that since the honeypot system in this paper will respond to the real PLC, it will decrease when the temperature drops. The value of Conpot is divided into a fixed value and a randomly generated value in the set value range, so there will be no numerical change corresponding to the external environment. It can be seen that the interactivity is low, so Conpot will become an invalid trap, making it difficult for attackers to take the bait.

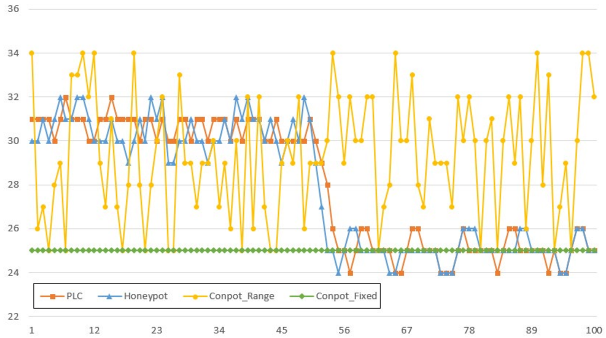

Figure 3 The Difference in Specific Register Value

4.2. *Shodan Honey Score Evaluation*

The Shodan scan results for the proposed honeypot system show that the honeypot is recognized as an ICS device and can adjust the exposed equipment information on port 502. Furthermore, the honeypot is awarded a honeyscore of 0.3 with the advisory message that the system appears to be real.

Shodan was also used to scan the Conpot IP. The results presented again indicate that the honeypot is an ICS device. However, in this case, the device specification is fully captured and the honeyscore has a value of 1.0. In other words, Shodan is fully confident that Conpot is a honeypot device.

5. Conclusion

This study has developed a honeypot description file framework for imitating the Modbus/TCP communication behavior of ICS equipment. The system is centrally controlled by a single honeypot controller and enables the deployment of multiple agents in different industrial environments, with more than one honeypot at each agent. The experimental results have shown that, in contrast to the Conpot honeypot, the proposed honeypot system accurately reproduces the response of a real-world PLC when processing Modbus function code 03 requests. Moreover, the honeypot successfully deceives the inspection mechanism of the Shodan IoT search engine and receives a honeyscore of just 0.3.

By contrast, the honeypot system proposed in the present study provides both good interactivity and a better ability to evade detection. As such, it potentially provides a more effective detection mechanism for real-world ICS networks.

Acknowledgements

This work was supported by the Delta Electronics, Inc.' Industry and Academic cooperation project - Design and Prototype of Hybrid Industrial Control System Honeypot and the Ministry of Science and Technology (MOST) in Taiwan under contract numbers MOST 110-2218-E-006-013-MBK and MOST 111-2218-E-006-010-MBK.

References

1. K. E. Hemsley, R. E. Fisher, History of Industrial Control System Cyber Incidents, Idaho Falls: Idaho National Laboratory, 2018.
2. Modbus Organization, "Modbus_Application_Protocol_V1_1b3," 2021. [Online]. Available: https://modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf. [Accessed Jun. 01, 2021].
3. Wikipedia, "JSON (JavaScript Object Notation), " [Online]. Available: https://en.wikipedia.org/wiki/JSON. [Accessed May 26, 2021].
4. A. Jicha, M. Patton, H. Chen, "SCADA honeypots: An in-depth analysis of Conpot," 2016 IEEE Conference on Intelligence and Security Informatics (ISI), Sep. 28-30, 2016.
5. R. Bodenheim, J. Butts, S. Dunlap, B. Mullins, "Evaluation of the ability of the Shodan search engine to identify Internet-facing industrial control devices," International Journal of Critical Infrastructure Protection, Vol. 7, No. 2, pp. 114-123, 2014.
6. SHODAN, "Honeypot Or Not?" 2021. [Online]. Available. [Accessed May 27, 2021].
7. R. Langner, "Stuxnet: Dissecting a Cyberwarfare Weapon," in IEEE Security & Privacy, Vol. 9, No. 3, pp. 49-51, 2011.
8. V. Pothamsetty, M. Franz., "SCADA HoneyNet Project: Building Honeypots for Industrial Networks," 2004. [Online]. Available. [Accessed May 27, 2021].

Authors Introduction

Dr. I-Hsien Liu

He is a research fellow in the Taiwan Information Security Center @ National Cheng Kung University (TWISC@NCKU) and Department of Electrical Engineering, National Cheng Kung University, Taiwan. He obtained his PhD in 2015 in Computer and Communication Engineering from the National Cheng Kung University. His interests are Cyber-Security, Wireless Network, Group Communication and Reliable Transmission.

Mr. Jun-Hao Lin

He received his B.S. degree from the Department of Electrical Engineering, National Taipei University of Technology, Taiwan in 2019. He got the M.S. degree in National Cheng Kung University in Taiwan. His research focuses on network communication and cyber security.

Mr. Hsin-Yu Lai

He received his B.S. degree from the Department of Electrical Engineering, National Chung Cheng University, Taiwan in 2020. He is acquiring the master's degree in Department of Electrical Engineering / Institute of Computer and Communication Engineering, National Cheng Kung University in Taiwan.

Dr. Jung-Shian Li

He is a full professor in the department of electrical engineering, National Cheng Kung University, Taiwan. He graduated from the National Taiwan University, Taiwan, with BS in 1990 and MS degrees in 1992 in electrical engineering. He obtained his PhD in 1999 in computer science from the Technical University of Berlin, Germany. He teaches communication courses and his research interests include wired and wireless network protocol design, network security, and network management. He is currently involved in funded research projects dealing with optical network, VANET, Cloud security and resource allocation, and IP QoS architectures. He is the director of Taiwan Information Security Center @ National Cheng Kung University. He serves on the editorial boards of the International Journal of Communication Systems.