Computers, Materials & Continua CMC, vol.65, no.2, pp.1723-1737, 2020

Identifying Honeypots from ICS Devices Using Lightweight Fuzzy Testing

Yanbin Sun[1], Xiaojun Pan[1], Chao Xu[2], Penggang Sun[2], Quanlong Guan[3], Mohan Li[1,*] and Men Han[4]

Abstract: The security issues of industrial control systems (ICSs) have become increasingly prevalent. As an important part of ICS security, honeypots and anti-honeypots have become the focus of offensive and defensive confrontation. However, research on ICS honeypots still lacks breakthroughs, and it is difficult to simulate real ICS devices perfectly. In this paper, we studied ICS honeypots to identify and address their weaknesses. First, an intelligent honeypot identification framework is proposed, based on which feature data type requirements and feature data acquisition for honeypot identification is studied. Inspired by vulnerability mining, we propose a feature acquisition approach based on lightweight fuzz testing, which utilizes the differences in error handling between the ICS device and the ICS honeypot. By combining the proposed method with common feature acquisition approaches, the integrated feature data can be obtained. The experimental results show that the feature data acquired is effective for honeypot identification.

Keywords: ICS, device, honeypot, identification.

1 Introduction

With the development of industrial technology, information technology (IT) and operation technology (OT) are merging in factory applications. The combination of IT and OT turns industrial control systems (ICSs) from closed to open. ICS devices (PLC, RTU, etc.) are usually connected to the Internet such that the ICS device can be operated remotely. These devices support richer information interaction, which enhances their function and brings convenience to the operator and the manager.

Although the utility of an ICS is significantly improved by connecting to the Internet, the security of an ICS is seriously affected. Since traditional ICSs are not subject to external

[1] Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou, 510006, China.

[2] Guangzhou Information Technology Security Evaluation Center, Guangzhou, 510006, China.

[3] Jinan University, Guangzhou, 510006, China.

[4] Kennesaw State University, 1100 South Marietta Pkwy Marietta, Georgia, 30060, USA.

[*] Corresponding Author: Mohan Li. Email: limohan@gzhu.edu.cn.

Received: 26 May 2020; Accepted: 24 June 2020.

security threats, their system software, application software and communication protocols are designed without considering a secure requirement, which introduces a great number of vulnerabilities. Meanwhile, ICSs are closely related to industrial production that requires continuity, i.e., 7 days×24 hours production without interruption. Thus, ICS devices are difficult to frequently stop and update for vulnerability mitigation. Current open ICSs do not solve the above problems but have all of the above security risks, which makes the security issue more serious.

To ensure the development of industry 4.0, ICS security has become a demand that must be examined. Proactive prevention is a promising approach for ICS security. It predicts and prevents potential security threats by proactively collecting and analyzing information from multiple threats. To protect ICS devices in a proactive manner, discovering and analyzing such devices on the Internet, as well as identifying potential security vulnerabilities, should be first completed. Developing ICS device scanning methods is the proper technology for the first step.

ICS device scanning is an efficient research for discovering ICS devices and locating device vulnerabilities. Scanning adopts two conventional methods (fingerprinting and banner grabbing) to discover devices. Based on the results, device vulnerabilities can be obtained according to vulnerability databases. Some mature systems (or tools) already exist for Internet-wide device scanning, such as Nmap, Shodan, Censys, Zoomeye, Oshadan, Ditecting, and so on. These systems also support ICS device discovery. However, the scanning technology has two sides. On the one hand, the technology can be used to identify security threats and prevent potential attacks. On the other hand, the technology can also be used maliciously by adversaries to launch attacks against the ICS, in which case ICS honeypots can be adopted to prevent malicious ICS device scanning.

ICS honeypots mostly simulate an ICS device to collect intelligence about adversarial motives and methods targeting the ICS. Honeypots are always deployed by a factory or research institution. They are used to detect attacks and provide intrusion alarms for factories. Honeypots can also be used to track and study attacks against ICSs. For example, the adversary may use a 0-day vulnerability to launch an attack against an ICS honeypot. By analyzing the attack process recorded in the ICS honeypot log, the 0-day vulnerability may be found by a researcher.

An ICS honeypot helps resist malicious ICS device scanning and a following attack. However, ICS honeypots face the following two issues: (1) For normal ICS device scanning, the honeypot is always incorrectly identified as a real device and provides fake data, resulting in inaccurate scanning results. (2) For the adversary, the honeypot may fail (i.e., it is identified as a fake device) because it cannot fully simulate the characteristics of a real device. Therefore, research for identifying honeypots rather than online ICS devices is essential. Some intelligent techniques, such as deep learning (DL) and machine learning (ML), may be promising solutions for honeypot identification. However, the main challenge for using intelligent techniques is the data, including what data to obtain and how to obtain it.

In this paper, we proposed an ICS honeypot identification framework based on which feature data acquisition is used to acquire the effective feature data for honeypot identification and is studied. Inspired by vulnerability mining, a lightweight fuzzy

testing-based feature acquisition approach is proposed, which finds the differences in error handling between an ICS device and honeypots by ensuring the security of the ICS device. Combining lightweight fuzzy testing with some common feature acquisition approaches, the integrated feature data can be obtained. Experiments show that the feature data works well for identifying an ICS honeypot.

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 describes the ICS device scanning. Section 4 presents feature acquisition using common approaches and the lightweight fuzzy testing approach. In Section 5, we analyze and evaluate the effectiveness of our approach. We then conclude this paper in Section 6.

2 Related works

With the development of network technology, security threats have been studied from the Internet to the Internet of Things (IoTs) [Li, Sun, Lu et al. (2020); Yin, Luo, Zhu et al. (2019); Wang, Chen, Song et al. (2018)], the Internet of Vehicles (IoVs) [Tian, Gao, Su et al. (2020)], the mobile Internet [Gu, Sun, Du et al. (2018)], and industrial control networks [Wan, Yao, Jing et al. (2018)]. Security issues for the Internet, IoTs and IoVs, e.g., security attacks [Tian, Luo, Qiu et al. (2019)], are widely studied. A variety of techniques, such as attack detection [Tian, Shi, Wang et al. (2019)], access control [Qiu, Tian, Du et al. (2020)], encryption [Xue, Yu, Li et al. (2019); Min, Yang, Wang et al. (2019); Long, Peng and Li (2018)], and signatures [Gu, Jia and Zhang (2017)] have been proposed. Online ICSs also face various security threats. An important aspect of security research for online ICS devices is security scanning, which has attracted much attention. Identification and anti-identification are important aspects of security scanning. The following work examines three parts: ICS scanning techniques, ICS honeypot techniques and an ICS honeypot identification method.

Scanning is an active technology used to detect security threats. Network scanning technology has been developed for many years and is relatively mature. At present, there are some powerful Internet-wide scanning platforms, such as NMap, Shodan, Censys, Zoomeye, Oshadan, and so on. These platforms support all networked devices, including industrial devices. There are also various tools specifically designed for networked industrial equipment scanning, such as PLCScan and Modscan.

Scanning approaches for ICS devices using the above tools (platforms) are similar. The corresponding port is first detected, then communication is carried out using industrial control protocols based on the open port. Since the interaction using industrial control protocols commonly relies on control commands, the scanner can obtain ICS device information through various control commands. Based on the information, there are two types of methods to identify the device including banner grabbing [Durumeric, Adrian, Mirian et al. (2015)] and fingerprinting [Shamsi, Cline and Loguinov (2017)]. The former is more suitable for ICS devices than the latter. On the one hand, the information obtained by these tools always includes the slave ID, device name, firmware version, CPU number, and so on, which can be used to identify the device. On the other hand, the limited number of training data and large number of device types make fingerprinting infeasible. However, the limitation of banner grabbing is that the banner information needs to be associated with the device artificially, and the scalability and efficiency are

affected, an issue that has received previous attention [Feng, Li, Wang et al. (2018)]. After a device is scanned, the scanning tool can be further used for vulnerability detection [Antrobus, Frey, Green et al. (2016)] and security situation analysis [Bano, Richter, Javed et al. (2018)].

In contrast, honeypots are a passive technology for security threat detection. By analyzing security logs, attack behavior and the source of attacks can be traced.

However, scanning can be exploited by malicious actors to identify bypassing honeypots. There has always been confrontation between scanning and honeypots. Honeypots have been used for many years, but most honeypot technologies focus on the Internet rather ICS. Recently, with the increase in industrial safety requirements, industrial honeypots have gradually gained more attention.

An ICS honeypot is a physical device or a simulated device that is used to capture attacks against ICS. Since ICS devices are difficult to be virtualized, physical devices are expensive and cannot be deployed on a large scale. Thus, most ICS honeypots are simulated devices. The simulated device is essentially a program that simulates the interaction process of a physical device. An ICS honeypot is similar to a software defined ICS device, such as Open PLC [Alves, Buratto, Souza et al. (2014)]. In fact, there is a fundamental difference between the two types of software, and their goals are different. Software defined ICS devices focus on how to apply the device to the production process, while ICS honeypots pay more attention to security considerations.

According to honeypot interactivity, ICS honeypots are divided into the following three types [Lau, Klick, Arndt et al. (2016)]: low interactivity, medium interactivity and high interactivity. Conpot [Rist, Vestergaard, Haslinger et al. (2013)], which is a low interactivity honeypot, simulates a Siemens S7-200 PLC and supports Modbus, S7Comm, HTTP and SNMP protocols. This type of honeypot is widely deployed on the Internet. However, the Conpot has obvious fingerprint characteristics and is easily identified by adversaries. CryPLH [Buza, Juh´asz, Miru et al. (2014)] simulates a Siemens Simatic 300(1) PLC device and supports data reading, SNMP and HTTP protocols. It is a medium interactivity honeypot. However, the OS fingerprint of CryPLH is different from a real device, and it can be easily identified by scanning tools.

Xpot [Lau, Klick, Arndt et al. (2016)] simulates Siemens S7-300 series PLCs. It is a high-interactive honeypot and supports PLC program compilation and interpretation, as well as network stack simulation of PLC. For seasoned adversaries, Xpot can still be identified.

Honeypot and anti-honeypot methods for honeypot identification appeared after the honeypots came into existence. Both adversaries and honeypot researchers have researched these methods. Differently, the former focuses on how to use the honeypot, and the latter focuses on how to bypass the honeypot. To ensure the effectiveness of the honeypot, the honeypot should remain active and be continuously improved to act against suspicious adversaries who attempt to distinguish between honeypots and real devices.

The approaches for identifying traditional honeypots have been under development for decades. However, ICS honeypots have recently received attention and there are a few ICS honeypot identification studies. Shodan supported honeypot identification. The characteristics of known honeypots are extracted to identify honeypots. However,

functions are not special for ICS honeypots, and an unknown or updated honeypot may be not identified. Fingerprinting of ICS honeypots [Feng, Li, Wang et al. (2016)] can also be used for honeypot identification, but this approach relies on the number of fingerprints, which is difficult to obtain. Our work does not rely on fingerprinting; however, we propose a framework and use a new feature error handling and corresponding lightweight fuzzy testing process to identify a honeypot.

In addition to the logical characteristic, the physical characteristics can also be used for identification, such as a wireless multimedia device identification system using radio frequency fingerprinting [Zhang, Li, Wang et al. (2018)], activity recognition using channel state information [Li, Xu, Li et al. (2019)], and inherent equipment distortion-based 3D source identification [Peng, Yang, Li et al. (2019)]. For this research, some artificial intelligence methods are adopted to identify devices, activities and sources, which can also be combined with the features obtained in our method.

3 ICS device scanning

To identify honeypots, the process of ICS device scanning should be first reviewed to better understand honeypot identification. Existing ICS device scanning is based on ICS protocols. Different from the traditional network standard protocol, the data transmission of ICS devices and software widely uses special, diverse protocols. There is no common standard for a large number of ICS protocols, which are divided into standard protocols (such as Modbus, DNP3, Ethernet/IP, etc.), private public protocols (such as FINS, Melsec, etc.) and private protocols (such as S7, PPI, etc.).

The architecture and process of ICS device scanning are shown in Fig. 1. Three databases are used for scanning including (1) the IP database, which contains the active IP addresses for scanning, (2) the protocol database, which contains the form of protocol and supports the protocol scanning, and (3) the feature database, which contains the features (most are ICS device banners) of different device types.

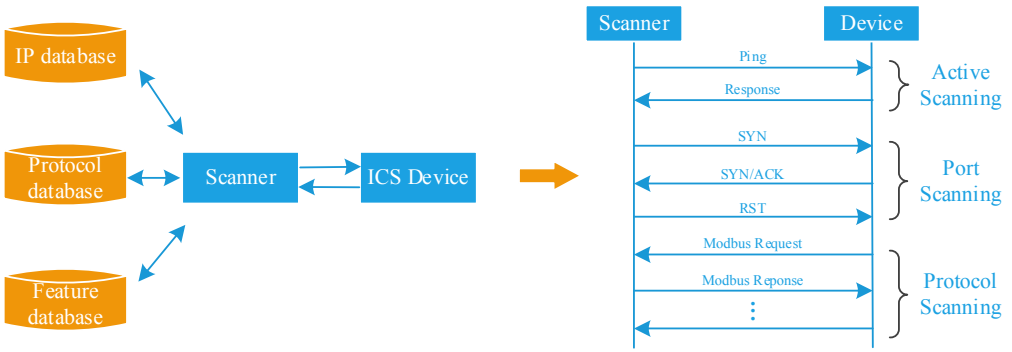

Figure 1: Architecture and process of ICS device scanning

The scanning process for ICS devices can be divided into the following three steps: (1) Active scanning is used to determine whether the device is alive. Devices in the entire network or a given target network segment can be scanned by sending ICMP packets (Ping), TCP SYN packets, etc.; the IP of the surviving device is stored in the IP database.

(2) Port scanning is used to scan the active device for a specific port from the IP database. Specifically, the open device port can be determined by sending TCP SYN packets, TCP FIN packets, etc. Based on the obtained port state, the available protocol of the device can be obtained. (3) Protocol scanning is used to identify a device based on the specific ICS protocol, which is key in identifying ICS devices.

The characteristics of ICS protocol are that the message structure is relatively fixed and the interaction process is mostly a simple request-response mode. For the interaction process, the request message includes the function code and the request content, as well as the device ID, length, and so on. The response message also includes the corresponding function code, as well as a response content to the request. For the ICS packet, there exist some special function codes that enable the requester to read the device configuration, state and memory. ICS device identification is based on these special function codes to construct and send request packets. After receiving the response data, the interaction process can be repeated using different function codes according to requirements. The received data may include ICS device description, firmware version, etc. Then, the ICS device can be identified by comparing the received data with the banner information in the feature database.

4 ICS honeypot identification

In this section, we focus on the feature acquisition. Some common feature acquisition approaches for ICS device identification are proposed and analyzed. To solve the shortcomings of common approaches, a lightweight fuzzy testing-based feature acquisition approach is then proposed.

4.1 Common features and acquisition approaches

ICS honeypots cannot simulate all the behaviors and features of an ICS device, so there must be some differences between an ICS honeypot and a real device. In this subsection and the next subsection, we determine which features are selected and how to acquire these features such that ICS honeypots and the ICS device can be easily separated. Here, we propose and analyze some common features for ICS honeypot identification. The features for honeypot identification can be classified into two categories based on the way and difficulty of acquiring features, namely, the surface features and the deep features.

4.1.1 Surface features

Surface features, or the inherent information of the honeypot itself, are required for deployment. These features can be easily obtained with a few interactions with the ICS device. We can divide these features into the following categories:

(1) IP address. IP address represents the location of the ICS device in the network. The address is used to find the ISP or organizational information to which the device belongs, which can help us identify honeypots. If the IP address belongs to a cloud service ISP, such as Amazon, Alibaba, Google, etc., it can almost always be determined that the corresponding device is a honeypot. Meanwhile, some universities and research institutes may also deploy ICS honeypots for research purposes. Devices of these organizations

also have a certain probability of being honeypots. The mappings from IP address to ISP or organization can be obtained through some public IP databases, such as TaobaoIP, IP2asn, IP2region, or Geoip2. All of these databases can provide third-party interface to assist in querying the IP information. IP2ASN provides a free API for users to retrieve CIDR, AS number, AS description and country information of an IP address.

(2) Physical location. The physical location is the location of the ICS device in the real world. Based on the physical location, we can determine whether the ICS device is in a factory area. If so, it is likely to be a real device. Based on an IP database, the latitude and longitude of a device can be obtained according to its IP address, and then the real physical address of the device can be obtained using an electronic map.

(3) Open port. Unlike Internet devices, ICS devices have limited resources and are functionally specific, so an ICS device can only support a few network services. Moreover, ICS devices are mostly dedicated devices from different manufacturers with diverse types. ICS protocols only consider the design requirements of corresponding manufacturers, so the protocols are poorly standardized. Meanwhile, protocols of devices from different manufacturers are not compatible with each other. It is difficult for an ICS device to support nonstandard protocols of many different manufacturers at the same time. Therefore, we can use port statistics and port conflict rules as features for ICS honeypot identification. Port statistics are obtained by scanning the ports of a device. If the ICS device has too many open ports or multiple consecutive unrelated open ports, it is likely a honeypot. Port conflict rules can be formulated by collecting port conflict information or manufacturer incompatibility information. For example, port 9600 of the FINS protocol and port 102 of the S7 protocol belong to two different manufacturers, and they do not coexist on the same ICS device. If two conflicting ports of a device are open at the same time, the device is likely a honeypot.

4.1.2 Deep features

Surface features are obtained easily by some common approaches, but the features can only be used to identify some roughly deployed ICS honeypots. In this case, perhaps the deployer does not want to hide the honeypot at all. For a well-deployed honeypot, these features do not work well. In addition to the surface features, ICS honeypots have some deep features that need carefully constructed probe packets according to special protocol standards and deep mining based on response packets. The deep features include the following two main types: device fingerprints and interaction content. There are two main approaches to acquire the deep features including fingerprinting of the ICS device and high interaction.

Device fingerprinting adopts the traditional operating system fingerprint identification method. The method utilizes differences between different operating systems in implementing the TCP/IP protocol stack. By sending different types of probe data packets, the operating system fingerprint is obtained; then, the operating system is identified by comparison with the fingerprint database. ICS devices typically use embedded operating systems such as VxWorks, μLinux, and more. Therefore, an ICS device can be identified based on a comparison result. If the fingerprint is that of a nonembedded operating system, it is most likely not an ICS device. There are some

problems in device fingerprints, including the following: (1) There are many types of industrial control devices, and the fingerprint library is difficult to construct; (2) The accuracy of fingerprint recognition of the operating system directly affects the accuracy of honeypot identification, and operating system updates may cause failure of fingerprint methods; (3) Advanced honeypots may have the ability to mimic fingerprints, resulting in erroneous recognition results.

The high interaction-based feature acquisition approach utilizes the property that the ICS honeypot cannot simulate some advanced functions of an ICS device, such as reading the complete configuration and state, downloading, uploading or executing the PLC program, writing data to special memory, and so on. By constructing data messages with advanced function codes and data, the identification tool can interact with the ICS honeypot. According to the special behavior and results, the ICS honeypot can be identified. The shortcomings of this approach are obvious. High interactions, such as write operations, directly affect the running of the device. Performing deep interactions on real devices may affect real industrial production and cause unintended consequences. Therefore, identifying honeypots through high interaction is not feasible for ICS scanners. Correspondingly, low interaction behavior is easily learned by the honeypot, which invalidates the purpose of identifications with low interactions.

To avoid the shortcomings of the common feature acquisition approach and enhance identification, new feature acquisition is needed.

4.2 Lightweight fuzz testing

To solve the shortcomings of existing features, a lightweight fuzzy testing-based approach for feature acquisition is proposed.

The process of ICS honeypot identification is to find the difference between ICS devices and honeypots. Thus, we only need to find an approach that can find the difference and ensure that the affect on ICS devices is limited as much as possible. In the field of vulnerability mining, the security vulnerabilities of different types of software are not the same, and security vulnerabilities can be found through different inputs. Similarly, we can also input abnormal data to ICS honeypots and devices. By analyzing the error handling results, differences between the two types of devices are obtained.

Here, we propose an ICS honeypot feature acquisition approach based on fuzzy testing in reference to vulnerability mining [Yun, Lee, Xu et al. (2018)]. Fuzzy testing is a common method used to analyze the weaknesses of software and network protocols. Fuzzy testing can exploit the security vulnerabilities of a program or protocol by randomly constructing abnormal input data, which is generally constructed by the mutation approach or the generation approach. The mutation approach generates new abnormal inputs based on the basic input. The generation approach randomly generates some key parts based on the data structure and constructs a new input. Above all, fuzzy testing is essentially a process that utilizes the improper design of error handling of the program to discover security vulnerabilities.

The feature acquisition method based on lightweight fuzzy testing proposed in this paper belongs to the method for deep features, and it adopts the idea of vulnerability mining.

Differently, our purpose is to determine the differences between ICS devices and honeypots through error handling rather than exploiting the security vulnerabilities. ICS devices have already passed a variety of security vulnerability testing, such as code auditing, fuzzy testing, etc., during the design and production processes, and they have good anti-attack capabilities. Thus, ICS devices can be identified with different abnormal inputs. At present, research on ICS honeypots focuses on how to simulate the interaction process of devices to be more realistic and pays little attention to error handling. Meanwhile, due to a variety of errors, it is difficult to simulate error handling for ICS honeypots, and it is also challenging to cover all the errors. Thus, by analyzing the difference between ICS honeypots and ICS devices in error handling, ICS honeypots can be identified.

Different from fuzzy testing with aggressive test targets, fuzzy testing of our honeypot feature acquisition is lightweight and focuses on testing error handling. The target our approach is to construct abnormal input on the premise of safety operations of an ICS device and to analyze the behavior and results during error handling. Based on the extracted characteristics, an ICS device and honeypot can be distinguished. Here, we present the following four types of abnormal inputs:

- *Abnormal message structure*. According to the specific ICS protocol packet format, an abnormal network packet with an illegal structure can be constructed as an abnormal input. Taking the Modbus TCP protocol as an example, the Modbus packet contains two parts, ADU and PDU. Each part has a fixed format with multiple fields, and some fields have fixed sizes. The fixed format and fixed field can be randomly changed to obtain the abnormal input.
- *Abnormal associated data.* If two fields of a packet are associated, we can break the association to construct abnormal data. For example, as shown in Fig. 4, the value of the Modbus TCP length field is related to the length of the PDU. Thus, we can change the value of *len* or reduce/increase the length of the PDU to break the relevance between them.
- *Abnormal function code.* The function code is the basis of ICS industrial control protocol communication, which specifies the function to be performed by the device. The abnormal packet can be constructed using the function code that is not commonly used or is not within the legal range.
- *Abnormal access.* This type of input, such as access to the border, access to invalid addresses, etc., often leads to unpredictable results. It is one of the key areas of focus by error handling. Thus, we can construct a packet for abnormal access by modifying the legal address to the illegal address, e.g., the PLC coil has its address range. Then, we can construct packets to read the information out of the range and detect the error handling differences between ICS devices and honeypots.

To avoid affecting normal operation of an ICS device, the construction of abnormal input should follow the following principles:

- The length of field content changes should be kept constant or reduced, because incremental content changes may cause insecure problems, such as memory out of bounds.
- ICS device function to be tested should be limited to the function code with read

permissions.

- Memory write operations should be avoided, especially for memory out of bounds.
- The rate of fuzzy testing needs to be limited to reduce security threats to ICS devices.

To further ensure the security of a scanned ICS device, we can construct a secure fuzzy testing database with secure test cases or secure generation/variation rules by performing fuzzy testing on local ICS devices. For lightweight fuzzy testing, the test cases can be extracted from the database or generated according to various rules. Database construction is beyond the scope of this paper, and we only provide a feasible idea here.

5 Analysis and evaluation

The experiment evaluates the features acquired by the lightweight fuzzy testing approach. For feature acquisition, we mainly focus on the ISP (or organization) of the device, the open device ports, and the response data based on the lightweight fuzzy testing. The information of the ICS device is a list of IP addresses of the Modbus device provided by Oshodan. Due to the potential risks of fuzzy testing, we only perform small-scale scanning. During the experiment, we found that many ICS devices have short lifetimes, and more than 50% of the devices fail within a month.

Based on the existing IP set, Masscan [Graham (2013)] is used to scan port 502 to detect if it is open. For alive IP addresses, we queried the ISP or organization information from various IP databases, IP2ASN, Ip2regin, and Taobaoip, for improved accuracy. By combining the three query results, a final result is obtained. We divided the ISPs into the following three types: enterprise ISP, university (or institute) ISP and cloud ISP. Tab. 1 shows the percentage of each type of ISP. We can see that most ICS devices are deployed on enterprise ISPs, followed by university ISPs. Cloud ISPs host the lowest number of ICS devices.

Table 1: Distribution of ISPs

ISP	Enterprise	University	Cloud Platform
Percentage	85	13	2

Then, for alive hosts, open port scanning is performed. We focus on ports 1-1024, typical ICS ports such as 1089-1091, 1962, 2001, 4840, 5007, 10307, 10311, 10364, 10365, 18245, and 20547, and the top 20 most used ports.

The port status of ICS devices is then counted. Fig. 2 shows the percentage of open ports for each range. It can be seen that the most open port is 502, followed by port 80. Port 502 is the default port for the Modbus protocol, and port 80 provides web services for ICS devices. However, not all ICS devices open port 502, likely because packet loss occurred or some hosts are closed or hidden during port scanning. We further count the open port numbers of each device, and Fig. 3 shows the distribution of the results. Most ICS devices open fewer than 5 ports.

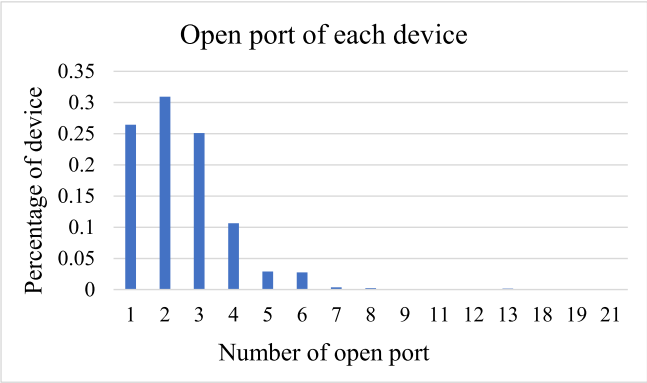

Figure 2: Distribution of open ports

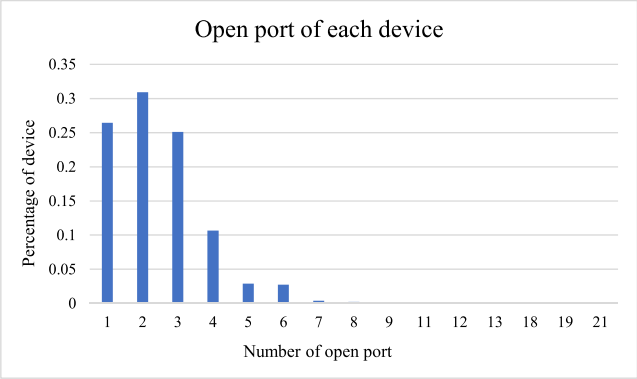

Figure 3: Distribution of open port numbers

Figs. 4-7 show the response data statistics of packets with different function codes. Due to the protection mechanism, most ICS devices do not directly respond to the requests. The main response includes two types, return a packet with an error code or do not respond to the request. As shown in Fig. 4, when the association in the packet is broken, the error handling of some devices is changed from a responding error packet to no response. However, some ICS devices still respond the normal packets even if the an error packet is returned. We found that IP addresses comes from cloud ISPs. Thus, we infer that these devices are ICS honeypots.

The results of the packet with the 05 function code and the packet with the 43 function code are similar to packets with the 01 function code. Some devices also return data corresponding to abnormal access addresses, including devices from the cloud ISP. For abnormal function code packets (Fig. 7), most devices respond to a packet with error code 01 (error01), but some devices respond to another packet with error code 01 (error01'). The last 4 bytes of error01 and error01' are "0301f102" and "03017101", respectively. Obviously, the latter packet is an abnormal error response. By analyzing the ISP information, it is found that devices with abnormal responses are distributed among three types of ISPs, and ICS honeypots are identified. Above all, it can be seen that the

error handling of ICS honeypots is very different from error handling in real ICS devices; therefore, this is an effective method for honeypot identification.

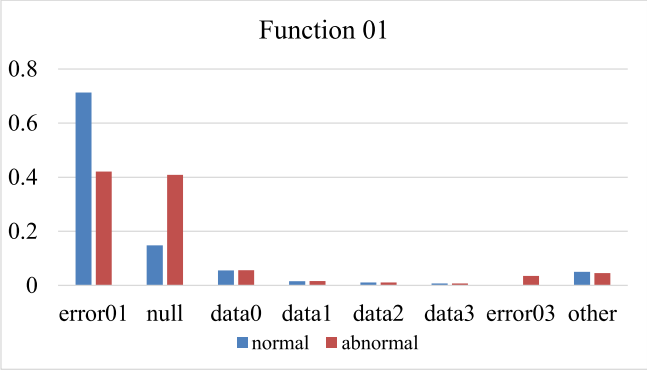

Figure 4: Response of function 01

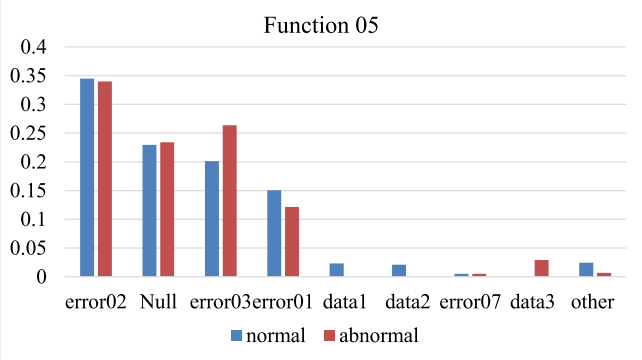

Figure 5: Response of function 05

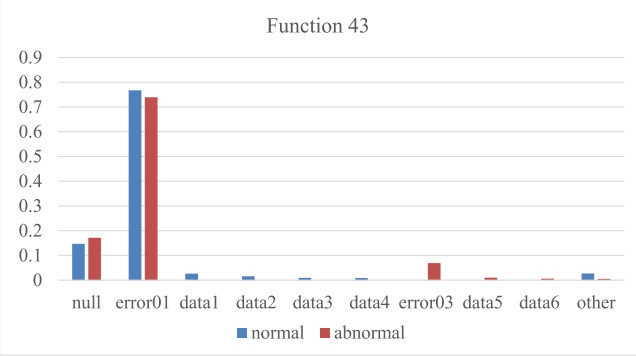

Figure 6: Response of function 43

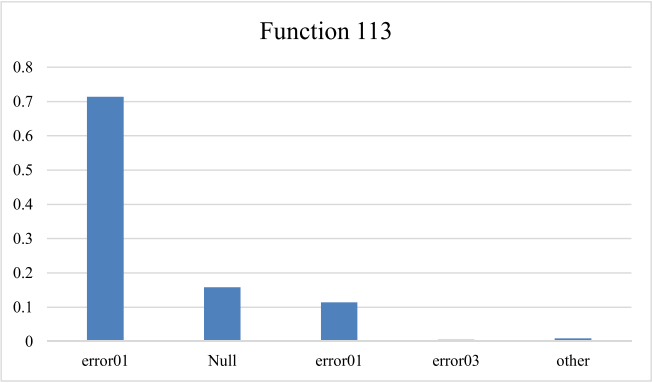

Figure 7: Response of function 113

5 Conclusion

This paper focuses on ICS honeypot identification and proposes a honeypot identification framework. Based on this framework, a method of acquiring feature data is studied. By analyzing an existing common feature acquisition approach, we proposed lightweight fuzzy testing drawing on the idea of vulnerability mining. The proposed approach ensures the security of an ICS device and acquires the feature data by analyzing the error handling results. The experiments reveal that lightweight fuzzy testing can provide sufficient, effective and flexible identification features, which can be used to identify honeypots effectively.

Funding Statement: This work is supported by the National Key Research and Development Plan (No. 2018YFB0803504), the National Natural Science Foundation of China (Nos. 61702223, 61702220, 61871140, 61872420, 61602210, U1636215), the Guangdong Province Key Area R&D Program of China (No. 2019B010137004, 2019B010136001), the Guangdong Province Universities and Colleges Pearl River Scholar Funded Scheme (2019), the Guangdong Basic and Applied Basic Research Foundation (2020A1515010450), the Science and Technology Planning Project of Guangdong (2017A040405029, 2018KTSCX016, 2019A050510024), the Science and Technology Planning Project of Guangzhou (201902010041), the Fundamental Research Funds for the Central Universities (21617408, 21619404) and the Opening Project of Shanghai Trusted Industrial Control Platform (TICPSH202003014-ZC).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Alves, T. R.; Buratto, M.; Souza, F. M.; Rodrigues, T. V. (2014): Openplc: an open source alternative to automation. *IEEE Global Humanitarian Technology Conference*, pp. 585-589.

Antrobus, R.; Frey, S.; Green, B.; Rashid, A. (2016): Simaticscan: towards a specialized vulnerability scanner for industrial control systems. *Proceedings of the 4th International Symposium for ICS & SCADA Cyber Security Research*, pp. 11-18.

Bano, S.; Richter, P.; Javed, M.; Sundaresan, S.; Durumeric, Z. et al. (2018): Scanning the internet for liveness. *ACM SIGCOMM Computer Communication Review*, vol. 48, no. 2, pp. 2-9.

Buza, D. I.; Juh´asz, F.; Miru, G.; Félegyházi, M.; Holczer, T. (2014): Cryplh: protecting smart energy systems from targeted attacks with a plc honeypot. *International Workshop on Smart Grid Security*, pp. 181-192.

Durumeric, Z.; Adrian, D.; Mirian, A.; Bailey, M.; Halderman, J. A. (2015): A search engine backed by internet-wide scanning. *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 542-553.

Feng, C.; Arshad, S.; Zhou, S.; Cao, D.; Liu, Y. (2019): Wi-Multi: a three-phase system for multiple human activity recognition with commercial WiFi devices. *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 7293-7304.

Feng, X.; Li, Q.; Wang, H.; Sun, L. (2016): Characterizing industrial control system devices on the internet. *IEEE 24th International Conference on Network Protocols*, pp. 1-10.

Feng, X.; Li, Q.; Wang, H.; Sun, L. (2018): Acquisitional rule-based engine for discovering internet-of-things devices. *Proceedings of the 27th USENIX Security Symposium*, pp. 327-341.

Graham, R. (2013): Masscan: the entire internet in 3 minutes. http://blog. erratasec.com /2013/09/masscan-entire-internet-in-3-minutes.html.

Gu, J.; Sun, B.; Du, X.; Wang, J.; Zhuang, Y. et al. (2018): Consortium blockchain-based malware detection in mobile devices. *IEEE Access*, vol. 6, no. 1, pp. 12118-12128.

Gu, K.; Jia, W. J.; Zhang, J. M. (2017): Identity-based multi-proxy signature scheme in the standard model. *Fundamenta Informaticae*, vol. 150, no. 2, pp. 179-210.

Lau, S.; Klick, J.; Arndt, S.; Roth, V. (2016): Towards highly interactive honeypots for industrial control systems. *ACM SIGSAC Conference on Computer and Communications Security*, pp. 1823-1825.

Li, M.; Sun, Y.; Lu, H.; Maharjan, S.; Tian, Z. (2020): Deep reinforcement learning for partially observable data poisoning attack in crowdsensing systems. *IEEE Internet of Things Journal*. https://doi.org/10.1109/JIOT.2019.2962914.

Long, M.; Peng, F.; Li, H. Y. (2018): Separable reversible data hiding and encryption for HEVC video. *Journal of Real-Time Image Processing*, vol. 14, no. 1, pp.171-182.

Min, Z.; Yang, G.; Wang, J.; Kim, G. J. (2019): A privacy-preserving BGN-type parallel homomorphic encryption algorithm based on LWE. *Journal of Internet Technology*, vol. 20, no. 7, pp. 2189-2200.

Peng, F.; Yang, J.; Lin, Z. X.; Long, M. (2019): Source identification of 3D printed objects based on inherent equipment distortion. *Computers & Security*, vol. 82, pp. 173-183.

Qiu, J.; Tian, Z.; Du, C.; Zuo, Q.; Su, S. et al. (2020): A survey on access control in the age of internet of things. *IEEE Internet of Things Journal*. https://doi.org/10.1109/JIOT.2020.2969326.

Rist, L.; Vestergaard, J.; Haslinger, D.; Pasquale, A. Smith, J. (2013): *Conpot ICS/SCADA Honeypot*. Honeynet Project.

Shamsi, Z.; Cline, D. B.; Loguinov, D. (2017): Faulds: a non-parametric iterative classifier for internet-wide OS fingerprinting. *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pp. 971-982.

Tian, Z.; Gao, X.; Su, S.; Qiu, J. (2020): Vcash: a novel reputation framework for identifying denial of traffic service in internet of connected vehicles. *IEEE Internet of Things Journal*. https://doi.org/10.1109/JIOT.2019.2951620.

Tian, Z.; Luo, C.; Qiu, J.; Du, X.; Guizani, M. (2019): A distributed deep learning system for web attack detection on edge devices. *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 1963-1971.

Tian, Z.; Shi, W.; Wang, Y.; Zhu, C.; Du, X. et al. (2019): Real time lateral movement detection based on evidence reasoning network for edge computing environment. *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4285-4294.

Wan, M.; Yao, J.; Jing, Y.; Jin, X. (2018): Event-based anomaly detection for non-public industrial communication protocols in SDN-based control systems. *Computers, Materials & Continua*, vol. 55, no. 3, pp. 447-463.

Wang, D.; Chen, D.; Song, B.; Guizani, N.; Yu, X. et al. (2018): From IoT to 5G I-IoT: the next generation IoT-based intelligent algorithms and 5G technologies. *IEEE Communications Magazine*, vol. 56, no.10, pp. 114-120.

Xue, L.; Yu, Y.; Li, Y.; Au, M. H.; Du, X. et al. (2019): Efficient attribute-based encryption with attribute revocation for assured data deletion. *Information Sciences*, vol. 479, pp. 640-650.

Yin, L.; Luo, X.; Zhu, C.; Wang, L.; Xu, Z. et al. (2019): Connspoiler: disrupting c&c communication of IoT-based botnet through fast detection of anomalous domain queries. *IEEE Transactions on Industrial Informatics*. https://doi.org/10.1109/TII.2019.2940742.

Yun, I.; Lee, S.; Xu, M.; Jang, Y.; Kim, T. (2018): QSYM: a practical concolic execution engine tailored for hybrid fuzzing. *Proceedings of the 27th USENIX Security Symposium*, pp. 745-761.

Zhang, Z.; Li, Y. B.; Wang, C.; Wang, M. Y.; Tu, Y. et al. (2018): An ensemble learning method for wireless multimedia device identification. *Security and Communication Networks*. https://doi.org/10.1155/2018/5264526.