IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON ADVANCES IN MACHINE LEARNING AND COGNITIVE COMPUTING FOR INDUSTRY APPLICATIONS

Received February 17, 2020, accepted February 24, 2020, date of publication February 27, 2020, date of current version May 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2976745

Machine Learning Methods for Industrial Protocol Security Analysis: Issues, Taxonomy, and Directions

JIAPING MEN[1], ZHUO LV[2], XIAOJUN ZHOU[3], ZHEN HAN[1], HEQUN XIAN[4], AND YA-NAN SONG[5]

[1]Beijing Key Laboratory of Security and Privacy in Intelligent Transportation, Beijing Jiaotong University, Beijing 100044, China
[2]State Grid Henan Electric Power Research Institute, Zhengzhou 450052, China
[3]Institute of Information Engineering, CAS, Beijing 100864, China
[4]College of Computer Science and Technology, Qingdao University, Qingdao 266071, China
[5]School of Business, Macau University of Science and Technology, Taipa, Macau

Corresponding author: Jiaping Men (15112066@bjtu.edu.cn)

This work was supported in part by the Natural Science Foundation of China under Grant U1736114, and in part by the National Key Research and Development Program of China under Grant 2017YFB0802805.

ABSTRACT Machine learning has been widely studied in the security analysis of Industrial Control Systems (ICSs). However, in industrial scenarios, the amount of data as well as the speed of data generation are very different from standard machine learning data sets. Using these heterogeneous data and finding meaningful insights for practical security applications in ICSs is a big challenge. In addition, ICSs have been built for quite a long time. Security has not been seriously taken into account when ICSs were built. Security assessment or attack prevention cannot always be done in real time, as an ICS requires to be online all the time, especially when it comes to systems that affect critical infrastructure. In this work, we are motivated to a provide a clear and comprehensive survey of the state-of-the-art work that employs machine learning in security applications in ICSs, including vulnerability analysis, vulnerability detection and exploitation, anomaly detection and security assessment. Based on our in-depth survey, we highlight the issues of industrial protocol analysis with machine learning methods, provide the security applications with machine learning in ICSs and indicate the future directions.

INDEX TERMS Protocol vulnerability, vulnerability analysis, machine learning, exploitation, ICS security.

I. INTRODUCTION

Industrial control systems have been widely used in various control domains, such as energy, municipal, water conservancy, railway, petroleum and petrochemical, etc., which are indispensable for the stable operation of the current industry. ICS protocols are the channels for the control system to transmit information, whose security are closely related to the safe and stable operation of the entire system. Therefore, this paper analyzes the security vulnerabilities of the ICS protocol and gives examples of exploits, aiming to attract the attention of industrial security researchers and take appropriate security measures in a timely manner before it is too late.

There are currently various ICS protocols in industrial control systems, while specific industries typically use only one or several specific protocols. Of course, for industries with complex processes, such as manufacturing, multiple protocols may be used, which makes the matter complicated. SCADA protocols include a series of fieldbus protocols that use serial link communication and several Ethernet protocols based Ethernet communication, including some application layer protocols over the TCP/IP protocol.

At the beginning of the development of industrial control systems, the protocols used were basically fieldbus protocols based on serial links, which mainly solved the digital communication between field devices such as intelligent instrumentation, controllers and actuators in industrial fields and information transmission issues between these field control devices and advanced control systems. Due to the outstanding advantages of fieldbus, such as easy-to-use, reliable, and economical, it has been highly valued by many standards bodies

The associate editor coordinating the review of this manuscript and approving it for publication was Min Xia.

and computer manufacturers, which lead to a prosperous development era.

Due to the wide variety of fieldbus and different standards, many people hope that Ethernet technology can intervene in the lower layers of the device and widely replace the existing fieldbus technology. Schneider is an active advocate and practitioner of this idea. A number of industrial products and practical applications have been available. With the advent and development of Ethernet technology and the advancement of Internet technology, early serial link-based fieldbus protocols gradually evolved to Ethernet-based. For example, Modbus protocol cluster has serial link based MODBUS RTU, MODBUS ASCII, MODBUS PLUS and Ethernet based MODBUS TCP; Profibus protocol cluster has serial link based PROFIBUS FMS, PROFIBUS DP, PROFIBUS PA and Ethernet based PROFINET CBA, PROFINET IO; IEC60870 protocol cluster has serial link based DNP3 and Ethernet based DNP3/UDP, DNP/TCP; IEC 60870-5-101 based on serial link and IEC60870-5-104 based on Ethernet. In addition to the above-mentioned several traditional general protocols, protocols widely used by ICS in the power industry include protocols such as ICCP (IEC 60870-6, TASE.2) and IEC 61850.

Due to the limitations of technology and the interests of various manufacturers, such a variety of industrial bus technologies coexist. The status of Ethernet technology will continue to penetrate for a while, but in any case, the Ethernet-based ICS protocol is still the trend of future development.

Enhancing the security of the ICS protocols is one of the important ways to enhance the overall security of the industrial control system. A basic analysis of the protocol will help expose the security issues that exist in the protocol, which in turn will guide the development of the security mechanism and eventually be incorporated into the protocol description. Due to the diversity of industrial control protocols and the existence of a large number of proprietary protocols, the security analysis of industrial protocols is relatively difficult task. Therefore, the traditional methods cannot be adopted directly. Fortunately, there is a large amount of research work using machine learning algorithms for security analysis, exploitation detection and security assessment of industrial protocols.

We make the following contributions:

- We demonstrate the reasons for the formation of vulnerabilities and combines experiments to achieve exploits for a number of protocols. It aims to provide a guide to the security enhancement of existing protocols or the design of new high security protocols by analyzing the security of the protocol and fully exposing the existing security risks.
- We provide the applications of machine learning algorithms on industrial protocols for security analysis, vulnerability exploitation detection, and security assessment. We also provide four methods to perform the assessment of the ICS protocols and make a comparative analysis of these methods.
- Based on our in-depth analysis of machine learning methods for industrial protocol security analysis, we summarize their issues and taxonomy, and provide a comprehensive perspective for future research work.

The structure of this paper is as follows. Section II sorts and categorizes relevant research work, Section III analyzes the causes of two types of vulnerabilities in ICS protocols and machine learning algorithms respectively, and Section IV analyzes possible attacks of ICS protocols, giving the attack implementation examples, and classifies the exploitation detection algorithms. Section V introduces some methods on how to make risk assessment on ICS protocols and gives a comparative analysis of different machine learning algorithms. Section VI concludes this paper.

II. RELATED WORK

Many researchers have conducted research on the security of ICS protocols, mainly focused on the following aspects: ICS protocols security analysis, ICS protocols vulnerability mining and protocols security hardening.

A. ICS PROTCOLS SECURITY ANALYSIS

Luswata *et al.* [1] conducted a penetration test on Modbus TCP and tested existing security countermeasures unique to ICS systems, giving some recommendations for improving ICS security. Grandgenett *et al.* [2] conducted exploit experiments on the CIP protocol, including exploits for authentication and privileged I/O in a CIP implementation, where CIP is an application-level protocol used by ICS components for communicating with each other. ANSI C12.22 specifies the communication interfaces for data communication networks in smart grids. Rrushi *et al.* [3] have identified several design vulnerabilities in the ANSI C12.22 protocol specification that can be exploited to cause denial of service attacks and service interruptions. They presented some revisions to the ANSI C12.22 protocol specification to mitigate the effects of these vulnerabilities. Yoo and Shon [4] developed a grammar-based fuzzing tool for extracting dynamic information from target program execution and experimenting it with two applications using the Modbus protocol, which is widely used in ICS systems. Samtani *et al.* [5] applied Shodan to search for ICS devices and evaluated their vulnerability by using Nessus tools against the National Vulnerability Database(NVD). Singh *et al.* [6] believed that DNP3 has exhaustive specification and is complex to implement, so they have specifically studied the attack on function code to help detect the security of protocol implementation.

Kim [7] discussed various forms of threats and vulnerabilities faced by IP-based wireless sensor networks and proposed a proper security management approach. Pidikiti *et al.* [8] analyzed the vulnerabilities of the IEC60870-5-101 & 104 communication protocols, which are widely used in power utilities sector, and conducted experiments on the vulnerability exploitation. They proposed an experimental model by using standard IEC62351 to mitigate

attacks. By analyzing the Modbus packets, Carcano *et al.* [9] inferred the correct behaviors of the ICS and discovered the critical state of the ICS, thus designing a state-based network intrusion detection system (NIDS). Liu *et al.* [10] pointed out that there are a large number of IP-based communication networks in substations, which are geographically dispersed, resulting in large attack surface for substations. They evaluated the threats facing ICS and gave some mitigation measures. Formby *et al.* [11] analyzed the security of TCP protocol, which used by many devices in the power grid as the transport layer of its application, and conducted experiments of TCP sequence number prediction attacks in the power grid. Cardenas et al. [12] conducted an attack on the process control system and discussed the risk assessment, detection and response of the industrial control system. Kalluri *et al.* [13] explained the possible vulnerabilities in the power grid, implemented a denial-of-service attack against the power grid, and gave an analysis of the impact of this attack. Bellettini and Rrushi [14] represented memory access taintedness as a decision tree to perform vulnerability analysis of ICS protocol binaries, aiming to mining memory corruption vulnerabilities in implementations. Cagalaban and Kim [15] employed the attack tree to model ICS and performed vulnerability analysis, and proposed a security framework to improve the security of ICS.

B. ICS PROTOCOLS VULNERABILITY MINING

Bratus *et al.* [16] designed an inline fuzzing test tool named LZFuzz, through the man-in-the-middle attack method, the two-way fuzzing test of ICS protocols was performed and achieved good results on ICS protocols that are proprietary or poorly documented. In our previous work [17], we designed a fuzzing tool called EUFuzz, aiming to solve the dilemma of poorly documented private protocols and low efficiency of fuzzing. EUFuzz can quickly identify the packet structure and guide the fuzzing process, which has achieved good performance. Choi *et al.* [18] proposed a multivariate static method to extract the protocol specification from the binary protocols used in ICS systems, thus using the protocol specification to guide fuzz testing.

C. ICS PROTOCOLS SECURITY HARDENING

Bagaria *et al.* [19] pointed out that the ICS legacy systems are inherently insecure, which utilize a large number of proprietary protocols, making the entire system extremely vulnerable to attacks. Therefore, they proposed a security-enhanced protocol version for DNP3, called Flexi-DNP3 (short for Flexible Distributed Network Protocol), which exchanges keys for data encryption during communication. By analyzing the DNP3 protocol, Graham and Patel [20] identified the threats and effective mitigation measures faced by ICS, and proposed the implementation of cost-effective countermeasures, including SSL/TLS, IPsec, encryption, and message authentication. Zhang *et al.* [21] proposed a series of lightweight anonymous mutual authentication with key agreement protocol on ECC and ARM Cortex-M0.

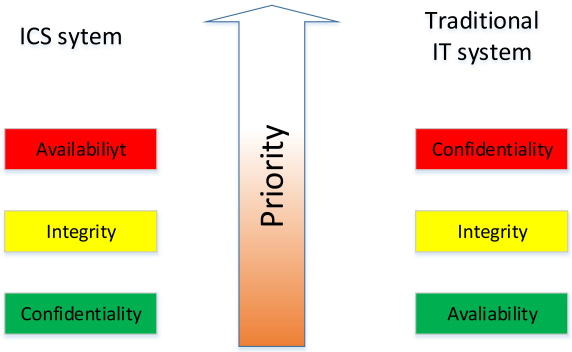


FIGURE 1. Differences between ICS and traditional IT systems.

There were also other research work related to ICS protocols. Lin *et al.* [22] applies Bro to the ICS system, constructs an intrusion detection system based on the DNP3 protocol, and defines the network events related to the protocol semantics. Xu *et al.* [23] provided a overview of recent advances in PLC attacks and protection technology, detailing some attacks scenarios. Zhou *et al.* [24] summarized the ICS security strategies of different countries from the perspective of ICS security standards. Related to software or system security, their exist work on the detection or analysis of anomalies or malware with static [25]–[33] or dynamic analysis [34]–[39] or with network traffic [40]–[51]. There also exist work on privacy analysis in smart devices [52], on secure protocols [53], [54] for authentication, or on IoT [55]. However, none of these work focuses on vulnerability analysis.

III. ICS PROTOCOL VULNERABILITY ANALYSIS

Identifying the causes of the protocol vulnerability is a prerequisite for our analysis of the vulnerability. After reviewing considerable literature, we have concluded that the industrial protocol vulnerabilities mainly due to two factors. First, the defects caused by the protocol designer [3], such as not considering the security dimension problem at the beginning of the design, or the logic flaw of the protocol specification itself, etc. Second, vulnerabilities introduced in the development process [2], [6], such as cross-border checks on some boundaries due to the skill level of the programmer.

A. VULNERABILITIES IN DESIGN

There exist big differences between industrial control systems and traditional IT systems shown in Fig. 1. ICS pays more attention to system availability and business continuity. Moreover, the early ICS is generally in a physical isolation situation, and the production environment is relatively closed, and people lacking professional knowledge generally cannot obtain the corresponding ICS attack and defense research environment and related system information. Therefore, at the beginning of the design of the ICS protocols, more consideration was given to the real-time, efficiency and convenience of the protocol, and designers did not consider

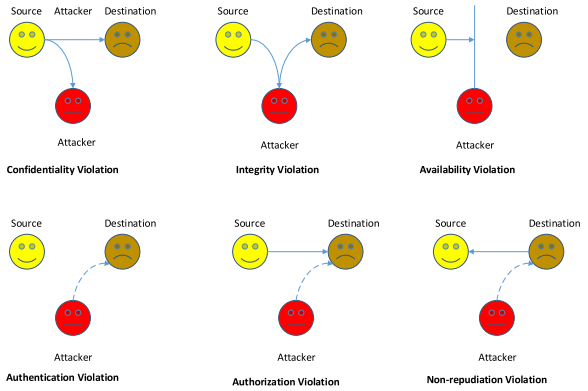


FIGURE 2. Security attributes ignored by early ICS protocol designers.

the related network security risks. They believed that all network communication and communication subjects within the ICS were legal, thus ignoring the basic security attributes of network communication(as shown in Fig. 2). However, as ICS heavily used IP-based control protocols, or migrated legacy serial-based protocols to the application layer of TCP/IP, ICS began to be exposed to the Internet. The outbreak of the 2010 Stuxnet [56] attack marked the beginning of hackers' interest in ICS.

In terms of machine learning in protocol design vulnerabilities analysis, Comparetti *et al.* [57] used a variety of machine learning algorithms, including sequence alignment algorithm(the NeedlemanWunsch algorithm precisely), partitioning around medoids (PAM) clustering algorithm, Exbar algorithm, sk-strings algorithm, beams algorithm etc., to automatically infer the state machine of a protocol, which is a major improvement on the protocol reverse engineering. Furthermore, the authors themselves designed a state machine extraction algorithm, can extract messages of different types and generate a protocol specification containing a protocol state machine. Finally, the extracted protocol specification can be used as an input to a stateful fuzzing tool to discover security vulnerabilities of the specific protocol. Rrushi *et al.* [3]studied the design vulnerabilities of the ANSI C12.22 protocol used in smart grid, analyzed the architecture of the protocol, providing details of the vulnerabilities they identified, and conducted a series of exploit experiments to verify those vulnerabilities, and finally provided mitigation measures for these vulnerabilities. Caselli *et al.* [58] employed machine learning algorithms to automatically extract the BACnet protocol specification documents and convert them into intrusion detection rules to monitor network traffic, which can identify process control errors and potentially dangerous misconfigurations. Zhang *et al.* [59] carried out in-depth mining of the precise state machine of the protocol, and designed a protocol state machine space mining algorithm based on data packet queries. Through interactive syntax inference technology, it automatically learned to generate the protocol state machine. Shim *et al.* [60] studied the specification extraction of unknown protocols, using the

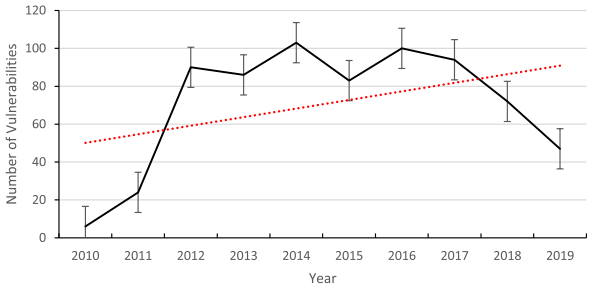


FIGURE 3. The trend of the number of vulnerabilities.

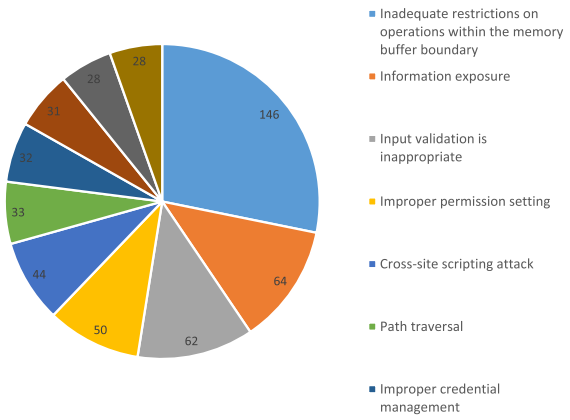


FIGURE 4. The number of vulnerabilities (counted by types).

Apriori-based CSP(Contiguous Sequence Pattern) machine learning algorithm to extract the protocol common strings, and using the tree structure-based CSP algorithm to extract the static fields of the protocol. It can extract all the static fields that are not used often but are possible. Lin *et al.* [22] designed and implemented an intrusion detection framework based on protocol specification, developed a new parser that supported the DNP3 protocol, analyzed the industrial control system traffic at runtime, extracted data from network packets, and verified compliance with the protocol specification.

B. VULNERABILITY IN IMPLEMENTATION

According to the vulnerability information reported by VLUHUB [61], the number of vulnerabilities in the ICS system is on the rise(Fig.3). Among them, according to the type of vulnerability, it is obvious that vulnerability in the design takes up only a small part, while the vulnerability introduced by the protocol implementation accounts for a large proportion(Fig.4).

Below we list some common insecure function calls and their solutions in C language, so that programmers can take proper checks on protocol implementation according to the protocol specification, so as to avoid making the same mistakes(Table 4).

Since the implementation of a specific protocol contains a lot of engineering details, which may not be completely consistent with the protocol specifications. It may lead to

TABLE 1. Insecure function calls and solutions in C Language.

Function	Severity	Solution
gets	highest risk	use fgts(buf,size,stdin) instead
strcpy	high rish	use strncpy instead
strcat	high risk	use strncat instead
sprintf	high risk	use snprintf instead, or use precision specifier
scanf	high risk	use precision specifier, or parsing by yourself
getchar/fgetc/ getc/read	intermediate risk	if used in a loop, be sure to check the buffer boundaries
fgetc	low rish	make sure the buffer size is as big as it is declared
fgets/memcpy/ snprintf/srccpy	high risk	use strncpy instead

the occurrence of vulnerabilities. Shu and Lee [62] tested the security of the protocol implementation using supervised machine learning methods, using Symbolic Parameterized Extended Finite State Machine (SP-EFSM) model to extract specific protocol, and then investigated the message confidentiality of the protocol implementation under the general Dolev-Yao attacker model [63]. Kim *et al.* [64] proposed a new fuzzing test case generation algorithm for the security test of protocol implementation in smart grid . According to the characteristics of the fuzzing test, the protocol fields were divided into three types, which can realize cross-domain and cross-layers test case generation. Zhao *et al.* [65] proposed a fuzzing test framework called SeqFuzzer, which used a deep learning model to automatically learn the protocol frame structure from communication traffic, processing the temporal features of the stateful protocol, and generating manipulated but seemingly reasonable messages as test cases. The EtherCAT protocol was tested and several vulnerabilities were detected. Niedermaier *et al.* [66] employed machine learning algorithms to learn the structure of proprietary protocols in industrial control system, and proposed a PropFuzz fuzzing test framework specifically for proprietary protocols. Huang *et al.* [67] used the improved differential evolution algorithm to improve the efficiency of fuzzing test of protocols in industrial control systems.

This Section analyzes the causes of vulnerabilities in the ICS protocol, including two main factors–vulnerability in design and vulnerability in implementation. A large number of machine learning algorithms are used to analyze these two types of problems, which have been introduced in the previous article. Table 3 provides the summary of this section.

TABLE 2. Machine learning based approaches for ICS protocol vulnerabilities analysis.

Literature	Algorithms	Protocol Design or Implementation
[57]	NeedlemanWunsch Algorithm, Partitioning Around Medoids (PAM) Clustering Algorithm, Exbar Algorithm, Sk-strings Algorithm, Beams Algorithm etc.	Protocol Design
[3]	Author-designed Algorithm	Protocol Design
[58]	Author-designed Algorithm	Protocol Design
[59]	Protocol State Machine Space Mining Algorithm	Protocol Design
[60]	Apriori-based CSP(Contiguous Sequence Pattern) Machine Learning Algorithm	Protocol Design
[22]	Author-designed Algorithm	Protocol Design
[62]	Supervised Machine Learning Methods	Protocol Implementation
[64]	Author-designed Algorithm	Protocol Implementation
[65]	Supervised Machine Learning deep learning model	Protocol Implementation
[66]	Author-designed Algorithm	Protocol Implementation
[67]	Improved Differential Evolution Algorithm	Protocol Implementation

This table provides a concise overview of what the focus of the available literature is. The next section is an experiment on the exploitation of some typical vulnerabilities.

IV. VULNERABILITY EXPLOITATION AND ANOMALY DETECTION

Rrushi [68] analyzed the vulnerabilities of the IEC61850 and Modbus protocols and showed how to exploit these vulnerabilities and maximize physical damage. Our work has big difference from his experiments. First, we enumerate the attack surface due to the ICS protocol vulnerability, that is, the possible attack entries and attack scenarios; then we exploit some of the typical vulnerabilities.

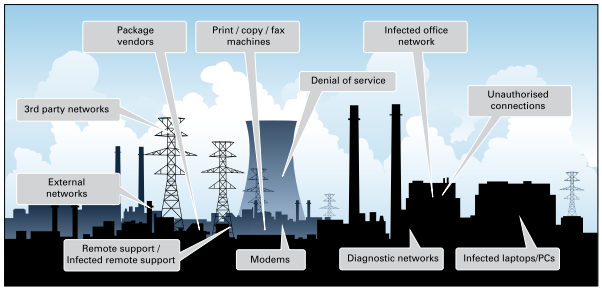


FIGURE 5. Potential points of entry of ICS system.

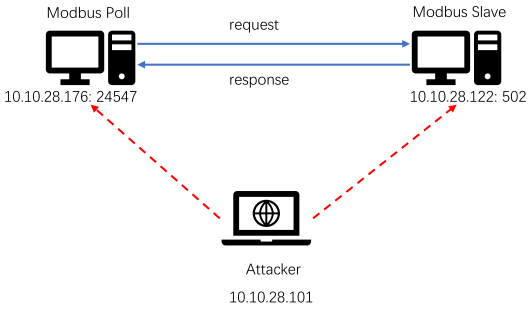


FIGURE 6. Experimental environment configuration.

A. POTENTIAL ATTACK SCENARIOS

Agostin [69] gave a detailed description of potential points of entry, which is shown in Fig.5

The potential attack scenarios include external networks, infected remote support, modems, diagnostic networks, infected laptops/PCs, unauthorized connections, infected office network, denial of service, print/copy/fax machines, package vendors and 3rd party networks. All of these potential attack scenarios have a lot to do with the protocols' vulnerabilities. As long as attackers can communicate with the ICS network, they can easily implement an attack.

B. PROOF OF CONCEPT

Now, let's conduct exploitation on ICS protocol vulnerabilities. Green *et al.* [70] employed the testbed environment to conduct man-in-the-middle attack experiments against ICS systems, showing the importance of understanding the process when conducting targeted ICS attacks. However, what if we have little knowledge of the operating mechanism of ICS, and we just employ the ICS protocol vulnerabilities, can we successfully implement a attack? The answer is 'yes'. Now we take Modbus as an example.

1) EXPERIMENT ENVIRONMENT

In order to minimize the destructiveness caused by actual attacks, we use Modbus simulation software [71] instead of real equipment, to conduct attack experiments to explain the existence and severity of vulnerabilities. The simulation suite includes two software-Modbus Poll and Modbus Slave. The experimental framework is shown in Fig.6.

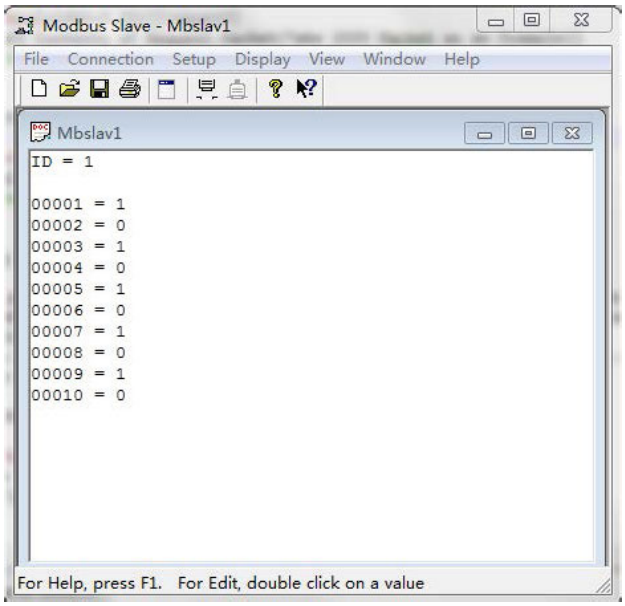


FIGURE 7. Normal communication of Modbus Slave.

We use two desktop computers to run Modbus Poll and Modbus Slave respectively, and one laptop to run the attack code. The basic configuration information of the desktop computer is as follows:

- Processor:Intel(R)Core(TM)i7-6700 CPU@3.4GHz
- Memory(RAM):16GB
- Operating System:64-bit Win7 Operating System Professional Service Pack1

The basic configuration information of the laptop is as follows:

- Processor:Intel(R)Core(TM)i7-7820HQ CPU@2.9GHz
- Memory(RAM):16GB
- Operating System:64-bit Win10 Operating System Home Version

In normal operation, Modbus Slave acts as a server for network communication, listening to port 502 and responding to data requests of Modbus Poll. Modbus Poll's port is randomly assigned by the operating system, in this experiment it is 24547. The initial status is Modbus Poll sending command to read the coil state, and Modbus Slave responding to it. As shown in Fig.7 & Fig.8, the polling cycle is 1000ms.

2) ATTACK PRACTICES

In this section, we launch four attack experiments in order to provide a detail information of communication information exposure, data tampering, illegal function code, and denial of service attack.

No.1. Communication Information Exposure:

Since the protocol does not have an encryption mechanism, all data communication processes can be monitored by a third party to infer some key information of the ICS system communication, such as the port used by the Master, the requested function code, the content of the response, and each coil's actual control state.

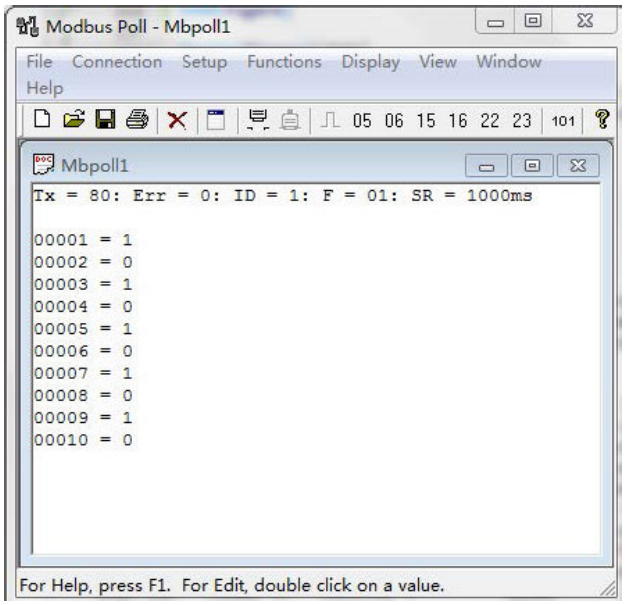


FIGURE 8. Normal communication of Modbus Master.

In this experiment, Wireshark [72] is used to capture network traffic. We can easily find out through the intercepted hands-shake packets(1578-1580)(Fig.9) that the port used by Modbus Master is 24547, whose ip address is 10.10.28.176, and the ip address of Modbus Slave is 10.10.258.122. Through the response packet(No.2333)(10) we can find that the function requested by Modbus Master is 'Read Coils', whose function code is 01H, and the base address of the request is 0, the requested number of the coils is 10, and the unit identifier of the peer Modbus Slave is 1. According to the response packet (No.2334)(11), we can find that the value of bit1, bit3, bit5, bit7, bit9 are all 1, and the rest coils' value is 0. Attackers can use these information to perform targeted strikes in a relatively precise manner, resulting in maximum physical damage.

No.2. Data Tampering Attack:

Since the Modbus protocol does not encrypt and authorize the function codes, we use the laptop to hijack the communication process and send 'Write Multi Coils' command, whose function code is 0FH, causing multi coils' states to change from 0 to 1. If it is in an actual ICS environment, this will cause a couple of switches to be in a closed state and even cause the entire ICS system to crash(Fig. 12).

No.3. Illegal Function Code:

Since the Mdbus protocol does not authorize the function code, we can easily forge the data requests. When the master and the Slave are communicating normally, we can use the laptop to send fake data packets containing illegal function code, causing the real Modbus Master to receive an erroneous response(Fig. 13).

No.4. DoS Attack:

Because Modbus lacks authentication, any machine can communicate with Modubs Slave via port 502. We use the laptop to continuously send data requests to 502 with a shorter polling time, 50ms for example, causing normal Modbus Master service interruption and receive nothing from the Slave(Fig. 14).

C. EXPLOITATION DETECTION

A lot of research has utilized machine learning algorithms to detect the vulnerability exploitation of ICS protocols. Schuster *et al.* [73] combined protocol-related knowledge and two unsupervised learning algorithms to complete attacks and fault detection in process control communication, to achieve self-learning anomaly detection. Beaver *et al.* [74] studied the application of multiple machine learning algorithms in detecting command and data injection attack scenarios, detected commands and controls in critical infrastructure facilities, and built models of benign and malicious command traffic to identify potential attack events. Anton *et al.* [75] employed anomaly detection algorithms based on machine learning and time series, and analyzed the traffic of industrial control networks using two different data sets to find the attack events, and compared the performance of SVM and random forest algorithms. Anton *et al.* [76] designed a time series-based anomaly detection method, i.e. the Matrix Profiles, to detect attacks in process data in an industrial environment, and compared the performance of the Matrix Profiles and one-class classifiers One-Class Support Vector Machines and Isolation Forest. Bernieri *et al.* [77] compared and evaluated the performance of anomaly detection machine learning algorithms in industrial control networks, and analyzed the advantages and limitations of two machine learning industrial network anomaly detection methods. Anton *et al.* [78] used machine learning-based anomaly detection algorithms to detect malicious traffic in Modbus/TCP communication traffic in virtual industrial scenarios. Supervised learning algorithms including support vector machines (SVMs), random forests, k-nearest neighbors (KNN), and k-means clustering were adopted. Through comparative analysis, SVM and KNN performed better on different data sets. Zolanvari *et al.* [79] introduced the application of machine learning based models in the Industrial Internet of Things Intrusion Detection System (IDS), and evaluated the performance of machine learning-based anomaly detection system in detecting these attacks against system deployment backdoors, command injection, and structured query language (SQL) injection etc. In response to the difficulty of anomaly detection in industrial control systems, Sokolov *et al.* [80] supplemented traditional anomaly detection methods and used machine learning-based anomaly detection methods, including the most common techniques for machine learning (decision trees, linear algorithms, support vector machines) and deep learning models (neural networks) and made comparative analysis of different performances Schuster *et al.* [81] utilized one-class SVM(OCSVM) to industrial control systems to detect anomalies in network traffic, which can be applied to the real-time environment with good performance. Wang *et al.* [82] reviewed the anomaly detection applications of machine learning in

1578	3.553791	10.10.28.176	10.10.28.122	TCP	66 24547 → 502 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
1579	3.553849	10.10.28.122	10.10.28.176	TCP	66 502 → 24547 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1580	3.555619	10.10.28.176	10.10.28.122	TCP	60 24547 → 502 [ACK] Seq=1 Ack=1 Win=65536 Len=0
1581	3.566570	10.10.28.176	10.10.28.122	Modbus/TCP	66 Query: Trans: 258; Unit: 1, Func: 1: Read Coils
1583	3.575986	10.10.28.122	10.10.28.176	Modbus/TCP	65 Response: Trans: 258; Unit: 1, Func: 1: Read Coils
1586	3.618342	10.10.28.176	10.10.28.122	TCP	60 24547 → 502 [ACK] Seq=13 Ack=12 Win=65536 Len=0
1904	4.592009	10.10.28.176	10.10.28.122	Modbus/TCP	66 Query: Trans: 259; Unit: 1, Func: 1: Read Coils
1905	4.605905	10.10.28.122	10.10.28.176	Modbus/TCP	65 Response: Trans: 259; Unit: 1, Func: 1: Read Coils
1939	4.648631	10.10.28.176	10.10.28.122	TCP	60 24547 → 502 [ACK] Seq=25 Ack=23 Win=65536 Len=0
2333	5.596882	10.10.28.176	10.10.28.122	Modbus/TCP	66 Query: Trans: 260; Unit: 1, Func: 1: Read Coils
2334	5.606743	10.10.28.122	10.10.28.176	Modbus/TCP	65 Response: Trans: 260; Unit: 1, Func: 1: Read Coils
2350	5.648850	10.10.28.176	10.10.28.122	TCP	60 24547 → 502 [ACK] Seq=37 Ack=34 Win=65536 Len=0
2668	6.612320	10.10.28.176	10.10.28.122	Modbus/TCP	66 Query: Trans: 261; Unit: 1, Func: 1: Read Coils
2674	6.622202	10.10.28.122	10.10.28.176	Modbus/TCP	65 Response: Trans: 261; Unit: 1, Func: 1: Read Coils
2679	6.664186	10.10.28.176	10.10.28.122	TCP	60 24547 → 502 [ACK] Seq=49 Ack=45 Win=65536 Len=0
3020	7.631495	10.10.28.176	10.10.28.122	Modbus/TCP	66 Query: Trans: 262; Unit: 1, Func: 1: Read Coils
3021	7.641685	10.10.28.122	10.10.28.176	Modbus/TCP	65 Response: Trans: 262; Unit: 1, Func: 1: Read Coils
3056	7.684205	10.10.28.176	10.10.28.122	TCP	60 24547 → 502 [ACK] Seq=61 Ack=56 Win=65536 Len=0
3267	8.644712	10.10.28.176	10.10.28.122	Modbus/TCP	66 Query: Trans: 263; Unit: 1, Func: 1: Read Coils
3273	8.654534	10.10.28.122	10.10.28.176	Modbus/TCP	65 Response: Trans: 263; Unit: 1, Func: 1: Read Coils
3308	8.697565	10.10.28.176	10.10.28.122	TCP	60 24547 → 502 [ACK] Seq=73 Ack=67 Win=65536 Len=0
3907	9.666941	10.10.28.176	10.10.28.122	Modbus/TCP	66 Query: Trans: 264; Unit: 1, Func: 1: Read Coils
3908	9.684533	10.10.28.122	10.10.28.176	Modbus/TCP	65 Response: Trans: 264; Unit: 1, Func: 1: Read Coils
3914	9.727731	10.10.28.176	10.10.28.122	TCP	60 24547 → 502 [ACK] Seq=85 Ack=78 Win=65536 Len=0

FIGURE 9. Handshake process when slave and master establish communication.

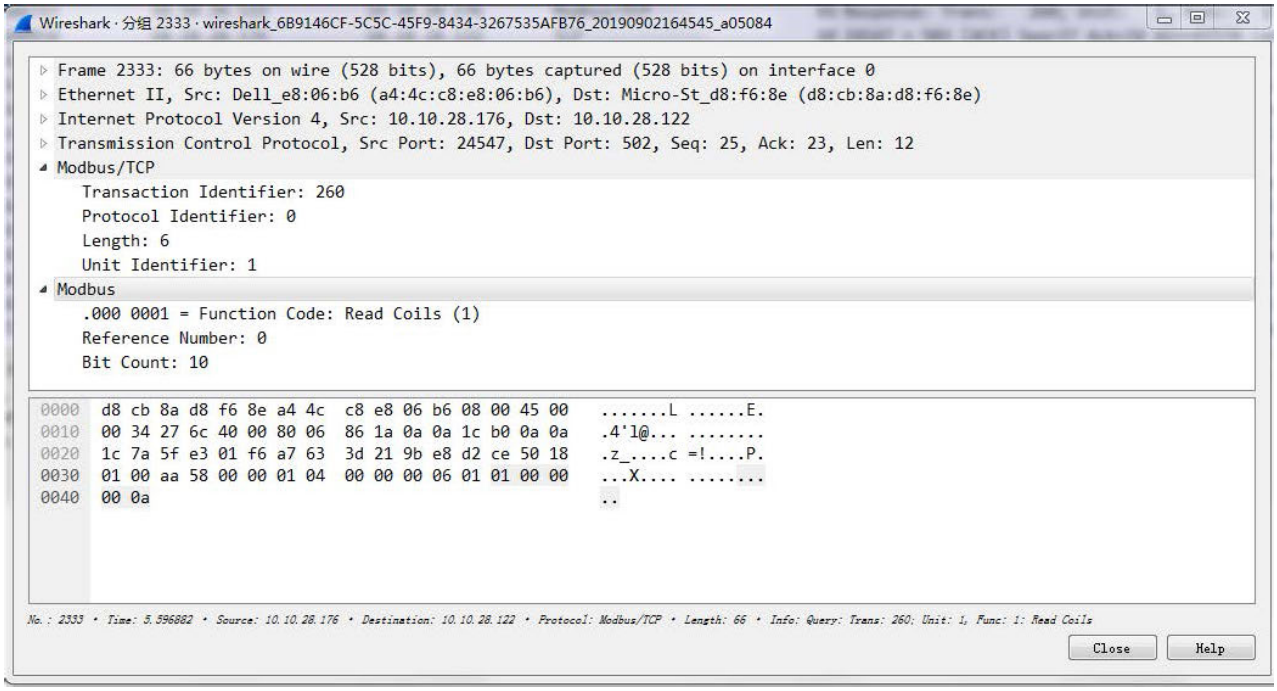


FIGURE 10. Detail contents of request packet (take 2333 packet as an example).

industrial networks, analyzed the advantages and disadvantages of different machine learning algorithms, and gave the future research trends of machine learning algorithms in the field of anomaly detection in industrial control system. Mantere *et al.* [83] analyzed the characteristics of network traffic in industrial control systems, and used machine learning-based methods to perform anomaly detection on network traffic in confined environments with good performance.

V. RISK ASSESSMENT

Attacks on ICS protocols are high-impact low-frequency (HILF) events [84], which means that we should make a detailed assessment instead of describing risk as "probability times consequence". Fortunately, there are some methods can be exploited to qualitatively or quantitatively make a risk assessment.

The risk assessment of the ICS protocols mainly includes three aspects [85]: the basic security elements, the analysis of threats, and the failure impact, as depicted in Fig.15.

A. RISK ASSESSMENT METHODS

1) FAULT TREE ANALYSIS

Fault Tree Analysis(FTA) [86] takes a top event failure of a system as the analysis target, finds the direct cause of failure, and decomposes from top to bottom, layer by layer, analyzes

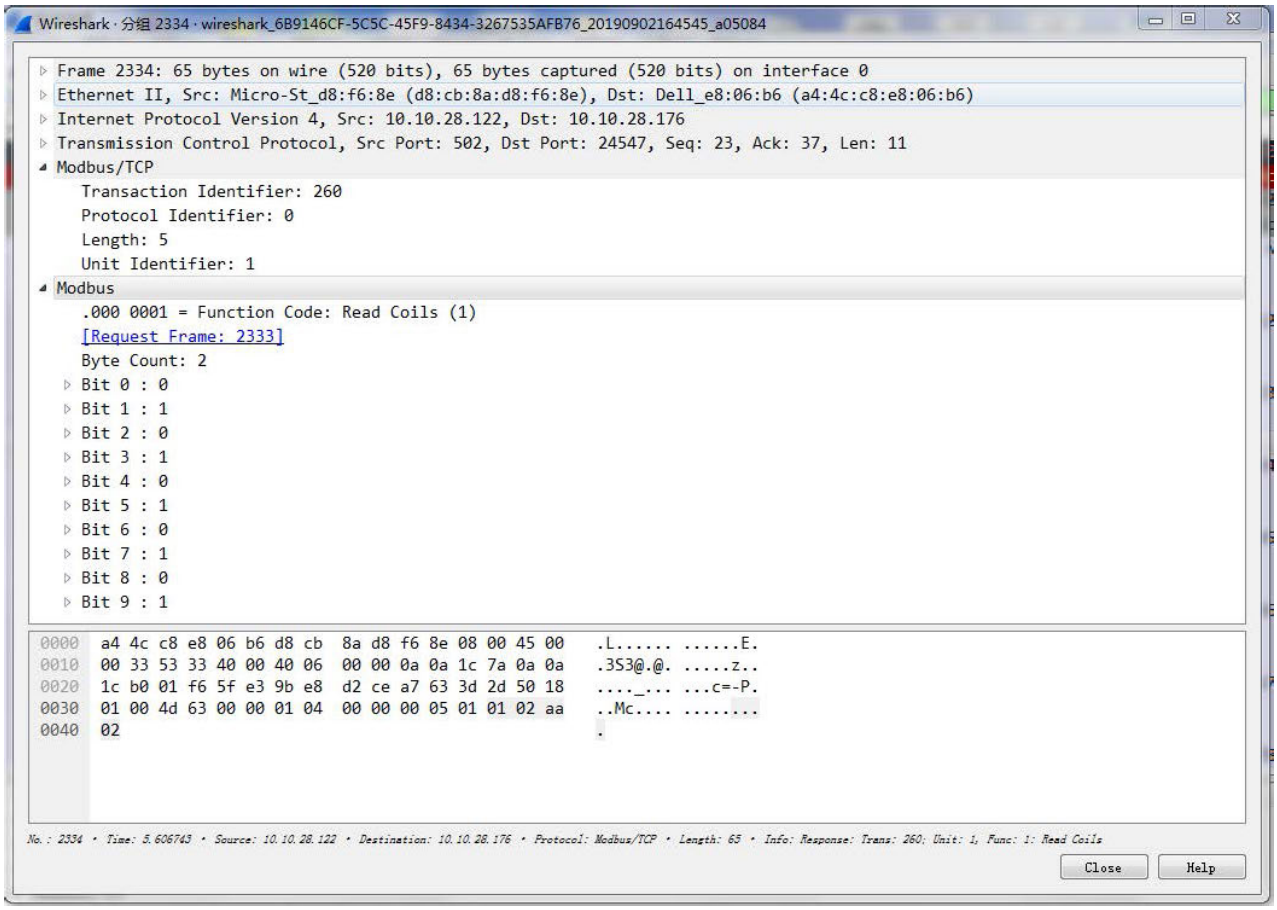


FIGURE 11. Detail contents of response packet (take 2334 packet as an example).

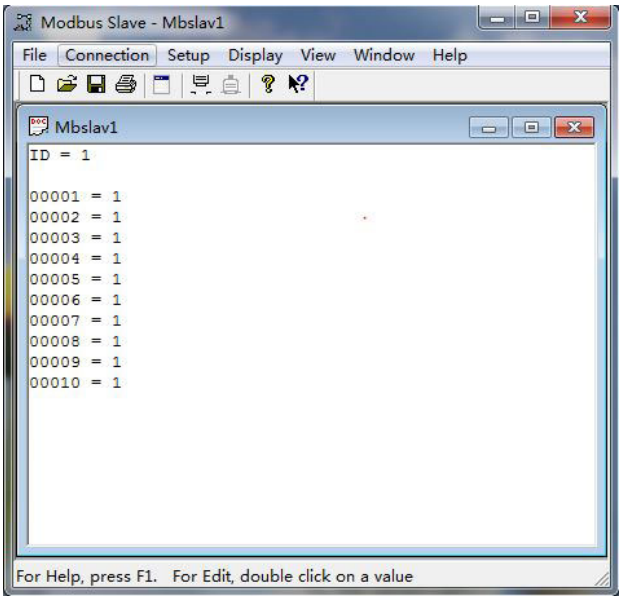


FIGURE 12. Data tampering attack scenario.

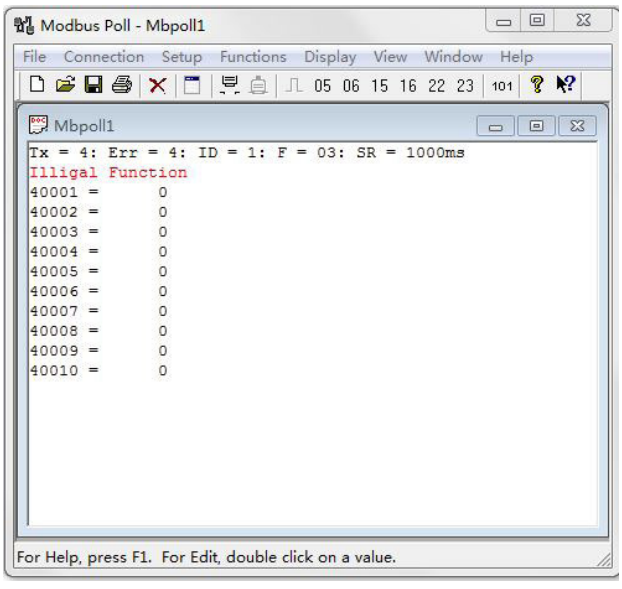


FIGURE 13. Illegal function attack scenario.

the cause and the probability of system failure through the logical relationship between intermediate event and bottom event and then takes preventive actions and design changes to achieve the desired purpose of the security of the protocol.

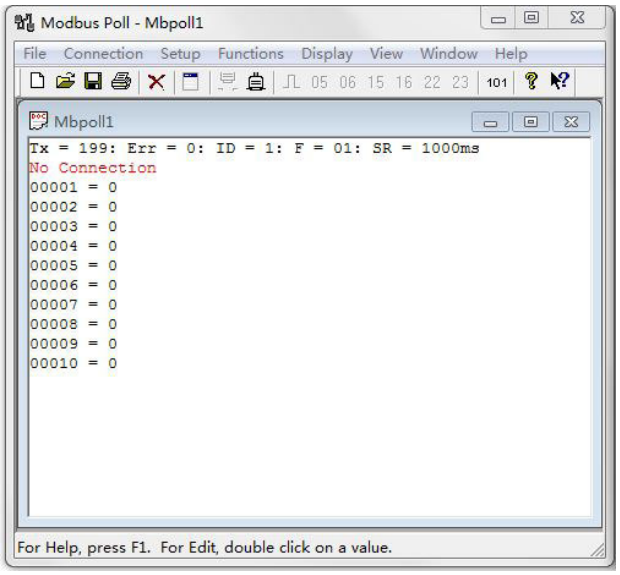


FIGURE 14. DoS attack scenario.

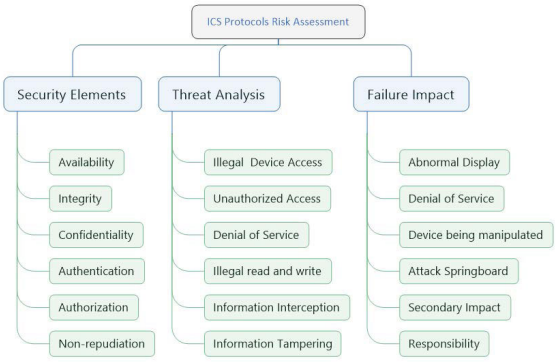


FIGURE 15. Content of risk assessment.

The fault tree is composed of many different event symbols and logic gates, and the logical relationship between events is represented by logic gates. These symbols can be divided into logical symbols, event symbols, and the like.

a: QUALITATIVE ANALYSIS OF FTA

Qualitative analysis aims to find failure modes that are possible and can cause the top event to occur, and to find all the minimum cut sets (MCS) of the fault [86], as shown in Fig.16. The simplified FTA provides a valuable basis for the designers and analysts to perform a qualitative analysis of the system. Even if the probability of the bottom event is not clear, the analysis of the MCS clearly tells administrator which areas are the weakest part of the system reliability.

b: QUANTITATIVE ANALYSIS OF FTA

Quantitative analysis is based on the probability of occurrence of the bottom event, with a certain degree of confidence to estimate the probability of occurrence of the top event, providing a quantitative basis for reliability design and analysis [86], as shown in Fig.17 and Fig.18.

In general, complex systems are very difficult to achieve the solution and usually requires simplification, so we use independent approximation, taking of all events as

TABLE 3. Machine learning based approaches for ICS protocol vulnerabilities analysis.

Literature	Algorithms	Purpose
[73]	One-class Support Vector Machine(OCSVM), Isolation Forest	Attack and Fault Detection
[74]	oneR, J48, RandomForest, Navive Bayes, SVM, and non-nested generalized exemplars(Nnge)	Overview
[75]	SVM, RandomForest	Anomaly-based Intrusion Detection
[76]	the Matrix Profiles, One-class Support Vector Machine(OCSVM), and Isolation Forest	Attack Detection
[77]	SVM, RandomForest, k-Nearest Neighbour(KNN), One-class Support Vector Machine(OCSVM), and Autoencoder(AE)	Overview
[78]	SVM, RandomForest, k-nearest neighbors(KNN), and k-means clustering	Overview
[79]	I Bayes,RandomForest, OneR, J48, Navive Bayes, SVM, and non-nested generalized exemplars(Nnge),deep belief networks(DBNs), and conditional deep belief network(CDBN), artificial neural network(ANN)	Overview, Vulnerability Analysis
[80]	decision trees, linear algorithms, SVM and deep learning models	Overview
[81]	One-class Support Vector Machine(OCSVM)	Anomaly Detection
[82]	Q learning, Dyna-Q, SVM, Logistic Regression, artificial neural network(ANN) and Autoencoder(AE)	Overview
[83]	self-organizing maps (SOM) approach	Anomaly Detection

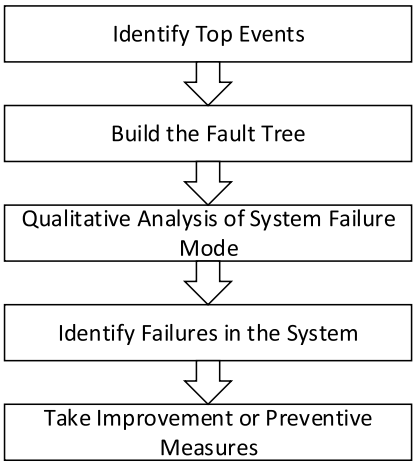


FIGURE 16. Qualitative analysis of FTA.

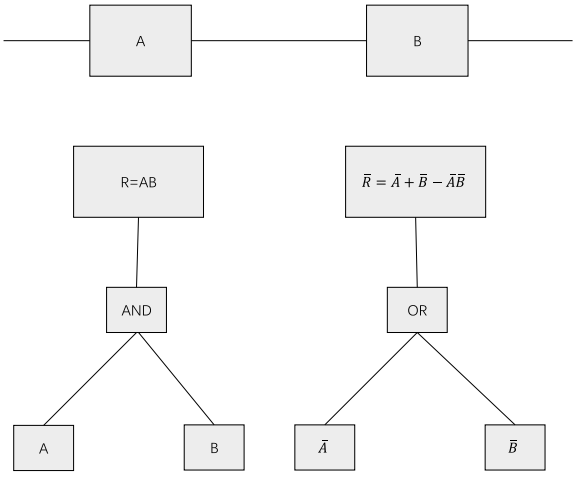


FIGURE 17. Quantitative analysis of FTA - series model.

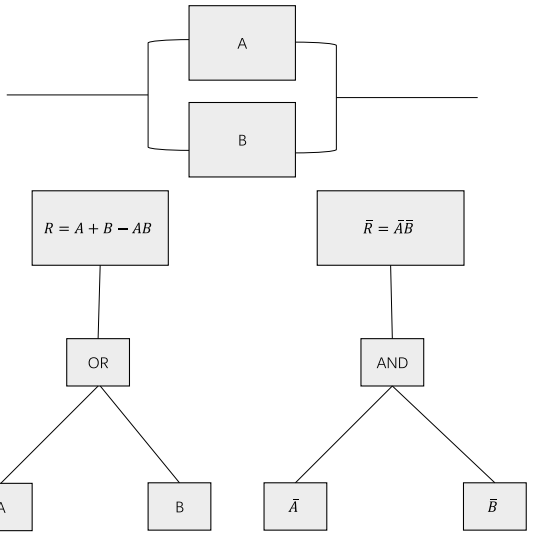


FIGURE 18. Quantitative analysis of FTA - parallel model.

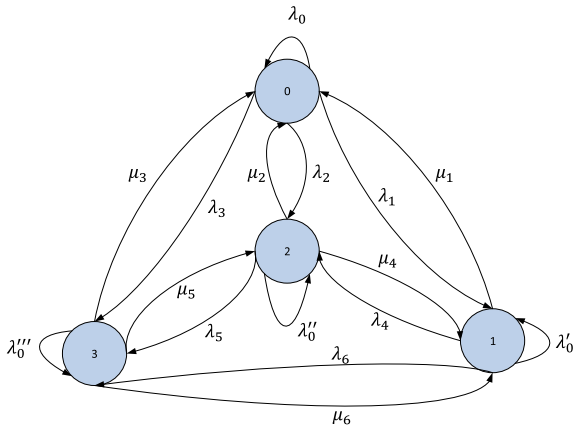


FIGURE 19. Markov model.

independent, and calculate the probability of the top event occurring from this assumption. There are two models, series model and parallel model. Fig.17 shows a series reliability model consisting of two components, A and B. The probability of success is $R = AB$, and the probability of system failure is $\overline{R} = 1 - AB = \overline{A} + \overline{B} - \overline{A}\,\overline{B}$. Fig18 shows a parallel reliability model consisting of two components, A and B. The probability of success is $R = 1 - \overline{A}\,\overline{B} = A + B - AB$, and the probability of system failure is $\overline{R} = \overline{A}\,\overline{B}$.

2) MARKOV MODEL

Markov Model [87] is a statistical model based on stochastic process theory. Its original model is the Markov Chain, which is used to study the state space migration of discrete event in dynamic systems. Markov chain is a Markov process whose time and state are both discrete, abbreviated as $X_n = X(n), n = 0, 1, 2 \ldots$

In the process, given the current knowledge or information, the past is irrelevant for predicting the future. At each step of the Markov chain, the system can change from one state to another according to the probability distribution, and can also maintain the current state. The state change is called a transfer.

The Markov chain is a sequence of random variables $X_1, X_2, X_3 \ldots$. The range of these variables, that is, the set of all their possible values, is called the "state space", and the value of Xn is the state of time n. If the conditional probability distribution of $Xn + 1$ for past states is only a function of X_n, then $P(X_{n+1} = x \mid X_0, X_{,1}, X_2, \ldots, X_n) = P(X_{n+1} = x \mid X_n)$ Here x is a state in the process. The above identity can be seen as a Markov property [88].

As shown in Fig.19, the circle indicates different states, the source of the state transition indicated by the arrow's starting point, the circle pointed by the arrow indicates the destination of the state transition, and the number on each arrow represents the probability of the state transitioning between states. A first-order process with M states has a squared state transition of M. The probability of each transition is called the state transition probability, which is the probability of moving from one state to another. The squared

probability of all of these M can be represented by a state transition matrix as following:

$$\begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{pmatrix} \lambda_0 & \lambda_1 & \lambda_2 & \lambda_3 \\ \mu_1 & \lambda_0' & \lambda_4 & \lambda_6 \\ \mu_2 & \mu_4 & \lambda_0'' & \lambda_5 \\ \mu_3 & \mu_6 & \mu_5 & \lambda_0''' \end{pmatrix} \end{array}$$

Using the Markov State Transfer Matrix, we can perform a quantitative risk assessment of the ICS protocols and the whole system.

3) BAYESIAN NETWORK

As mentioned above, the Markov chain describes a sequence of states. However, in many cases, the relationship between things cannot be chained together. At this time, the Bayesian network [89] is used: each state is only related to the state directly connected to it, and is not related to those indirectly connected to it. The topology of the Bayesian network is more flexible than the Markov chain, and is not constrained by the chain structure, which more accurately describes the correlation between events. The Markov chain is a special case of the Bayesian network, and the Bayesian network is a generalization of the Markov chain.

The Bayesian network, also known as the Belief Network, or the directed acyclic graphical model, which is one type of probability graph model. It is an uncertainty processing model that simulates causality in human reasoning. Its network topology is a directed acyclic graph (DAG). Let $G = (I, E)$ denote a DAG, where I represents a set of all nodes in the graph, and E represents a set of directed connected line segments, and let $X = (X_i), i \in I$, For a random variable represented by a node i in a DAG, if the joint probability of node X can be expressed as:

$$p(x) = \prod_{i \in I} p(x_i \mid x_{pa(i)})$$

Then X is called a Bayesian network relative to a directed acyclic graph G, where $pa(i)$ indicates the "cause" of node i, or the parent of node i.

For any random variable, the joint probability can be obtained by multiplying the respective local conditional probability:

$$p(x_1, \ldots x_k) = p(x_k \mid x_1, \ldots x_{k-1}) \ldots p(x_2 \mid x_1) p(x_1)$$

In fact, the Bayesian network can be seen as a nonlinear extension of the Markov chain. The significance of this feature is to clarify that the Bayesian network can easily calculate the joint probability distribution.

4) ATTACK TREE

The above three methods are concerned about failure scenarios, while attack tree [90], [91] is more concerned about malicious attempts to manipulate a system. The attack tree uses a tree structure to represent the attacks faced by the system, where the root node represents the target being attacked and the leaf node represents the method of achieving the attack target.

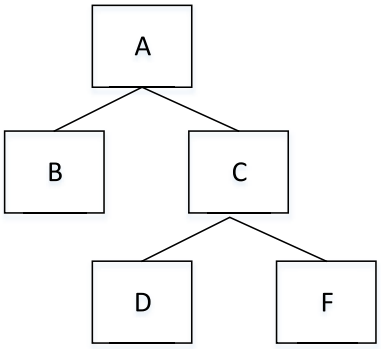


FIGURE 20. A simple example of attack tree.

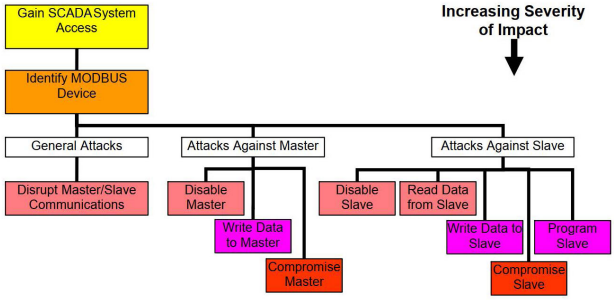


FIGURE 21. Using attack tree to assess ICS system.

The attack tree has multiple levels of nodes, including root and leaf nodes. The lower level of the root node is the leaf node, and the lower level of the leaf node is still the leaf node. For a leaf node, the lower-level leaf node directly drawn by it is its child node. Naturally, the leaf node is the parent node of its lower-level child node. For example, in Fig.20 below, we can see that the child nodes of leaf node C are leaf node D and leaf node F, then leaf node c is the parent node of leaf node D and leaf node F. In the attack tree, the child node must satisfy the condition that its parent node is true (i.e., node D can cause node C to be true).

Eric J. Byres [92] has utilized attack trees in assessing vulnerabilities in ICS system, which can be depicted in Figure 21.

B. METHODS COMPARISON

Different risk assessment methods have their own advantages and disadvantages and are applicable to different scenarios. Current mainstream risk assessment methods can be divided into two categories: methods based on knowledge reasoning and methods based on pattern recognition. Methods based on knowledge reasoning are the focus of current research and have achieved a lot of research results, which reduce the impact of the researchers' subjectivity on the risk assessment to a certain extent. However, this type of methods are less intelligent and are also limited by the formulation of inference rules and the acquisition of prior probabilities.

TABLE 4. Insecure function calls and solutions in C language.

Method	Category	Advantage	Disadvantage
FTA	Knowledge Reasoning	The cause-effect relationship is clear,can be used for both qualitative analysis and quantitative analysis and systematic evaluation	The analysis is local and the analyst must be very familiar with the object system being analyzed.
Markov Model	Pattern Recognition	The effect of the process state prediction is good, and it can be considered for the prediction of the dangerous state on the production site	Strict independence assumptions are required, and various probabilities of state changes are required. Knowledge about matrix operations is complicated.
Bayesian Network	Knowledge Reasoning	Performs well on small-scale data and is suitable for multi-classification tasks	Very sensitive to the representation of input data
Attack Tree	Knowledge Reasoning	A clear and organized way to describe the security threats the system faces and the multiple attacks the system may be subject to	When an attack tree is applied in a specific instance, its structure may become large and complex. A complete attack tree is likely to include hundreds or thousands of leaf nodes.

On the contrary, methods based on pattern recognition are more intelligent, but require a lot of training data to obtain the parameters of the model. A comparative analysis of the listed methods is performed below (See Table 4).

The current risk assessment methods have been applied in the actual security assessment process and have achieved good results. However, With the increasing complexity of the industrial control system and the continuous development of artificial intelligence technology, risk assessment need to be performed in a more intelligent way. The future trend is to use deep learning methods to build a knowledge map and perform correlation analysis based on the protocol vulnerabilities, so as to obtain global security situation awareness of the whole system.

VI. CONCLUSION

In this paper, we analyzed the causes of ICS protocol vulnerability and summarized the two main categories of vulnerabilities–vulnerability in design and vulnerability in implementation. We analyzed some potential attack scenarios and conducted experiments to exploit several vulnerabilities. In addition, we provided the applications of machine learning algorithms on industrial protocols for security analysis, vulnerability exploitation detection, and security assessment. We also provide four methods to perform the assessment of the ICS protocols and make a comparative analysis of these methods. Based on the research work presented in this paper, we conclude that in order to improve the security of the whole ICS system, protocol designers should consider security attributes when design an ICS protocol, whether it is public or proprietary; and protocol developers should conduct sufficient tests such as fuzzing-test before its implementation in practice; and system maintenance personnel should perform comprehensive risk assessment and take appropriate security measures timely. All these aspects of research work, machine learning methods can play a great role, and will play an increasingly important role in the field of industrial protocol security.

REFERENCES

[1] J. Luswata, P. Zavarsky, B. Swar, and D. Zvabva, "Analysis of SCADA security using penetration testing: A case study on modbus TCP protocol," in *Proc. 29th Biennial Symp. Commun. (BSC)*, Jun. 2018, pp. 1–5.
[2] R. Grandgenett, W. Mahoney, and R. Gandhi, "Authentication bypass and remote escalated I/O command attacks," in *Proc. 10th Annu. Cyber Inf. Secur. Res. Conf. (CISR)*, 2015, pp. 1–7.
[3] J. L. Rrushi, H. Farhangi, R. Nikolic, C. Howey, K. Carmichael, and A. A. Palizban, "By-design vulnerabilities in the ANSI C12. 22 protocol specification," in *Proc. 30th Annu. ACM Symp. Appl. Comput.*, 2015, pp. 2231–2236.
[4] H. Yoo and T. Shon, "Grammar-based adaptive fuzzing: Evaluation on SCADA modbus protocol," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Nov. 2016, pp. 557–563.
[5] S. Samtani, S. Yu, H. Zhu, M. Patton, and H. Chen, "Identifying SCADA vulnerabilities using passive and active vulnerability assessment techniques," in *Proc. IEEE Conf. Intell. Secur. Informat. (ISI)*, Sep. 2016, pp. 25–30.
[6] C. Singh, A. Nivangune, and M. Patwardhan, "Function code based vulnerability analysis of DNP3," in *Proc. IEEE Int. Conf. Adv. Netw. Telecommun. Syst. (ANTS)*, Nov. 2016, pp. 1–6.
[7] H. Kim, "Security and vulnerability of SCADA systems over IP-based wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 8, no. 11, Jan. 2012, Art. no. 268478.
[8] D. S. Pidikiti, R. Kalluri, R. K. S. Kumar, and B. S. Bindhumadhava, "SCADA communication protocols: Vulnerabilities, attacks and possible mitigations," *CSI Trans. ICT*, vol. 1, no. 2, pp. 135–141, Apr. 2013.

[9] A. Carcano, I. N. Fovino, M. Masera, and A. Trombetta, "State-based network intrusion detection systems for SCADA protocols: A proof of concept," in *Proc. Int. Workshop Crit. Inf. Infrastruct. Secur.* Springer, 2009, pp. 138–150.

[10] C.-C. Liu, C.-W. Ten, and M. Govindarasu, "Cybersecurity of SCADA systems: Vulnerability assessment and mitigation," in *Proc. IEEE/PES Power Syst. Conf. Expo.*, Mar. 2009, pp. 1–3.

[11] D. Formby, S. S. Jung, J. Copeland, and R. Beyah, "An empirical study of TCP vulnerabilities in critical power system devices," in *Proc. 2nd Workshop Smart Energy Grid Secur. (SEGS)*, 2014, pp. 39–44.

[12] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, "Attacks against process control systems: Risk assessment, detection, and response," in *Proc. 6th ACM Symp. Inf., Comput. Commun. Secur.*, 2011, pp. 355–366.

[13] R. Kalluri, L. Mahendra, R. K. S. Kumar, and G. L. G. Prasad, "Simulation and impact analysis of denial-of-service attacks on power SCADA," in *Proc. Nat. Power Syst. Conf. (NPSC)*, Dec. 2016, pp. 1–5.

[14] C. Bellettini and J. L. Rrushi, "Vulnerability analysis of SCADA protocol binaries through detection of memory access taintedness," in *Proc. IEEE SMC Inf. Assurance Secur. Workshop*, Jun. 2007, pp. 341–348.

[15] G. Cagalaban and S. Kim, "Towards Improving SCADA Control Systems Security with Vulnerability Analysis," in *Proc. Int. Conf. Parallel Distrib. Comput. Netw.* Springer, 2010, pp. 27–32.

[16] S. Bratus, A. Hansen, and A. Shubina, "LZfuzz: A fast compression-based fuzzer for poorly documented protocols," Darmouth College, Hanover, NH, USA, Tech. Rep. 634, 2008.

[17] J. Men, G. Xu, Z. Han, Z. Sun, X. Zhou, W. Lian, and X. Cheng, "Finding sands in the eyes: Vulnerabilities discovery in IoT with EUFuzzer on human machine interface," *IEEE Access*, vol. 7, pp. 103751–103759, 2019.

[18] S. Choi, Y. Chang, J.-H. Yun, and W. Kim, "Multivariate statistic approach to field specifications of binary protocols in SCADA system," in *Proc. Int. Workshop Inf. Secur. Appl.* Springer, 2014, pp. 345–357.

[19] S. Bagaria, S. B. Prabhakar, and Z. Saquib, "Flexi-DNP3: Flexible distributed network protocol version 3 (DNP3) for SCADA security," in *Proc. Int. Conf. Recent Trends Inf. Syst.*, Dec. 2011, pp. 293–296.

[20] J. H. Graham and S. C. Patel, "Security considerations in SCADA communication protocols," Tech. Rep., 2004.

[21] W. Zhang, D. Lin, H. Zhang, X. Zhou, Y. Gao, and C. Chen, "A lightweight multi-precision squaring on embedded processors for ECC," in *Proc. 17th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun./ 12th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE)*, Aug. 2018, pp. 1014–1019.

[22] H. Lin, A. Slagell, C. D. Martino, Z. Kalbarczyk, and A. K. Iyer, "Adapting Bro into SCADA: Building a specification-based intrusion detection system for the DNP3 protocol," in *Proc. 8th Annu. Cyber Secur. Inf. Intell. Res. Workshop*, 2013, p. 5.

[23] Z. Xu, X. Zhou, L. Wang, Z. Chen, K. Chen, Z. Yan, W. Zhang, and C. Chen, "Recent advances in PLC attack and protection technology," *J. Cyber Secur.*, vol. 4, no. 3, pp. 48–69, 2019.

[24] X. Zhou, Z. Xu, L. Wang, and K. Chen, "What should we do? A structured review of SCADA system cyber security standards," in *Proc. 4th Int. Conf. Control, Decis. Inf. Technol. (CoDIT)*, Apr. 2017, pp. 0605–0614.

[25] W. Wang, Z. Gao, M. Zhao, Y. Li, J. Liu, and X. Zhang, "DroidEnsemble: Detecting Android malicious applications with ensemble of string and structural static features," *IEEE Access*, vol. 6, pp. 31798–31807, 2018.

[26] G. Xu, B. Guo, C. Su, X. Zheng, K. Liang, D. S. Wong, and H. Wang, "Am i eclipsed? A smart detector of eclipse attacks for ethereum," *Comput. Secur.*, vol. 88, Jan. 2020, Art. no. 101604.

[27] W. Wang, J. Song, G. Xu, Y. Li, H. Wang, and C. Su, "ContractWard: Automated vulnerability detection models for ethereum smart contracts," *IEEE Trans. Netw. Sci. Eng.*, to be published.

[28] W. Wang, X. Wang, D. Feng, J. Liu, Z. Han, and X. Zhang, "Exploring permission-induced risk in Android applications for malicious application detection," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 11, pp. 1869–1882, Nov. 2014.

[29] X. Wang, W. Wang, Y. He, J. Liu, Z. Han, and X. Zhang, "Characterizing Android apps' behavior for effective detection of malapps at large scale," *Future Gener. Comput. Syst.*, vol. 75, pp. 30–45, Oct. 2017.

[30] W. Wang, Y. Li, X. Wang, J. Liu, and X. Zhang, "Detecting Android malicious apps and categorizing benign apps with ensemble of classifiers," *Future Gener. Comput. Syst.*, vol. 78, pp. 987–994, Jan. 2018.

[31] X. Liu, J. Liu, W. Wang, Y. He, and X. Zhang, "Discovering and understanding Android sensor usage behaviors with data flow analysis," *World Wide Web*, vol. 21, no. 1, pp. 105–126, Mar. 2017.

[32] W. Wang, M. Zhao, and J. Wang, "Effective Android malware detection with a hybrid model based on deep autoencoder and convolutional neural network," *J. Ambient Intell. Hum. Comput.*, vol. 10, no. 8, pp. 3035–3043, Apr. 2018.

[33] W. Meng, L. Jiang, K.-K.-R. Choo, Y. Wang, and C. Jiang, "Towards detection of juice filming charging attacks via supervised CPU usage analysis on smartphones," *Comput. Electr. Eng.*, vol. 78, pp. 230–241, Sep. 2019.

[34] W. Wang, M. Zhao, Z. Gao, G. Xu, H. Xian, Y. Li, and X. Zhang, "Constructing features for detecting Android malicious applications: Issues, taxonomy and directions," *IEEE Access*, vol. 7, pp. 67602–67631, 2019.

[35] X. Liu, J. Liu, S. Zhu, W. Wang, and X. Zhang, "Privacy risk analysis and mitigation of analytics libraries in the Android ecosystem," *IEEE Trans. Mobile Comput.*, to be published.

[36] W. Wang, X. H. Guan, and X. L. Zhang, "Modeling program behaviors by hidden Markov models for intrusion detection," in *Proc. Int. Conf. Mach. Learn. Cybern.*, vol. 5, Aug. 2004, pp. 2830–2835.

[37] W. Wang, X. Zhang, and S. Gombault, "Constructing attribute weights from computer audit data for effective intrusion detection," *J. Syst. Softw.*, vol. 82, no. 12, pp. 1974–1981, Dec. 2009.

[38] W. Wang, X. Guan, X. Zhang, and L. Yang, "Profiling program behavior for anomaly intrusion detection based on the transition and frequency property of computer audit data," *Comput. Secur.*, vol. 25, no. 7, pp. 539–550, Oct. 2006.

[39] L. Feng, W. Wang, L. Zhu, and Y. Zhang, "Predicting intrusion goal using dynamic Bayesian network with transfer probability estimation," *J. Netw. Comput. Appl.*, vol. 32, no. 3, pp. 721–732, May 2009.

[40] W. Wang, Y. Shang, Y. He, Y. Li, and J. Liu, "BotMark: Automated botnet detection with hybrid analysis of flow-based and graph-based traffic behaviors," *Inf. Sci.*, vol. 511, pp. 284–296, Feb. 2020.

[41] W. Wang, X. Guan, and X. Zhang, "Processing of massive audit data streams for real-time anomaly intrusion detection," *Comput. Commun.*, vol. 31, no. 1, pp. 58–72, Jan. 2008.

[42] X. Zhang, C. Furtlehner, and M. Sebag, "Data streaming with affinity propagation," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, Antwerp, Belgium, Sep. 2008, pp. 628–643.

[43] W. Wang and R. Battiti, "Identifying intrusions in computer networks with principal component analysis," in *Proc. 1st Int. Conf. Availability, Rel. Secur. (ARES)*, 2006, pp. 270–279.

[44] W. Wang, T. Guyet, R. Quiniou, M.-O. Cordier, F. Masseglia, and X. Zhang, "Autonomic intrusion detection: Adaptively detecting anomalies over unlabeled audit data streams in computer networks," *Knowl.-Based Syst.*, vol. 70, pp. 103–117, Nov. 2014.

[45] X. Zhang, C. Furtlehner, C. Germain-Renaud, and M. Sebag, "Data stream clustering with affinity propagation," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 7, pp. 1644–1656, Jul. 2014.

[46] W. Wang, J. Liu, G. Pitsilis, and X. Zhang, "Abstracting massive data for lightweight intrusion detection in computer networks," *Inf. Sci.*, vols. 433–434, pp. 417–430, Apr. 2018.

[47] L. Zhang, Z. Lv, X. Zhang, C. Chen, N. Li, Y. Li, and W. Wang, "A novel approach for traffic anomaly detection in power distributed control system and substation system," in *Proc. 13th Int. Conf. Netw. Syst. Secur.*, Sapporo, Japan, Dec. 2019, pp. 408–417.

[48] W. Wang, Y. He, J. Liu, and S. Gombault, "Constructing important features from massive network traffic for lightweight intrusion detection," *IET Inf. Secur.*, vol. 9, no. 6, pp. 374–379, Nov. 2015.

[49] X. Guan, W. Wang, and X. Zhang, "Fast intrusion detection based on a non-negative matrix factorization model," *J. Netw. Comput. Appl.*, vol. 32, no. 1, pp. 31–44, Jan. 2009.

[50] W. Wang, X. Guan, and X. Zhang, "A novel intrusion detection method based on principle component analysis in computer security," in *Proc. Int. Symp. Neural Netw.*, Dalian, China, Aug. 2004, pp. 657–662.

[51] W. Wang, X. Zhang, S. Gombault, and S. J. Knapskog, "Attribute normalization in network intrusion detection," in *Proc. 10th Int. Symp. Pervas. Syst., Algorithms, Netw.*, Kaohsiung, Taiwan, Dec. 2009, pp. 448–453.

[52] L. Li, J. Liu, L. Cheng, S. Qiu, W. Wang, X. Zhang, and Z. Zhang, "CreditCoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 7, pp. 2204–2220, Jul. 2018.

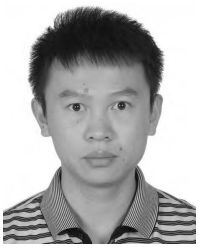
[53] G. Xu, J. Liu, Y. Lu, X. Zeng, Y. Zhang, and X. Li, "A novel efficient MAKA protocol with desynchronization for anonymous roaming service in global mobility networks," *J. Netw. Comput. Appl.*, vol. 107, pp. 83–92, Apr. 2018.
[54] X. Zeng, G. Xu, X. Zheng, Y. Xiang, and W. Zhou, "E-AUA: An efficient anonymous user authentication protocol for mobile IoT," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1506–1519, Apr. 2019.
[55] H. N. Dai, R. C. W. Wong, H. Wang, Z. Zheng, and A. V. Vasilakos, "Big data analytics for large-scale wireless networks: Challenges and opportunities," *ACM Comput. Surv.*, vol. 52, no. 5, p. 99, 2019.
[56] A. Thabet, "Stuxnet malware analysis paper," Code Project, 2011.
[57] P. M. Comparetti, G. Wondracek, C. Kruegel, and E. Kirda, "Prospex: Protocol specification extraction," in *Proc. 30th IEEE Symp. Secur. Privacy*, May 2009, pp. 110–125.
[58] M. Caselli, E. Zambon, J. Amann, R. Sommer, and F. Kargl, "Specification mining for intrusion detection in networked control systems," in *Proc. 25th USENIX Secur. Symp.*, 2016, pp. 791–806.
[59] Z. Zhang, Q.-Y. Wen, and W. Tang, "Mining protocol state machines by interactive grammar inference," in *Proc. 3rd Int. Conf. Digit. Manuf. Autom.*, Jul. 2012, pp. 524–527.
[60] K.-S. Shim, Y.-H. Goo, M.-S. Lee, H. Hasanova, and M.-S. Kim, "Inference of network unknown protocol structure using CSP(Contiguous Sequence Pattern) algorithm based on tree structure," in *Proc. IEEE/IFIP Netw. Oper. Manage. Symp.*, Apr. 2018, pp. 1–4.
[61] Vulhub Development Team. *Ics Security Information*. [Online]. Available: http://cve.scap.org.cn/view/ics
[62] G. Shu and D. Lee, "Testing security properties of protocol implementations-a machine learning based approach," in *Proc. 27th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2007, p. 25.
[63] I. Cervesato, "The Dolev-Yao intruder is the most powerful attacker," in *Proc. 16th Annu. Symp. Log. Comput. Sci.*, vol. 1, 2001, pp. 1–2.
[64] S. Kim, W. Jo, and T. Shon, "A novel vulnerability analysis approach to generate fuzzing test case in industrial control systems," in *Proc. IEEE Inf. Technol., Netw., Electron. Autom. Control Conf.*, May 2016, pp. 566–570.
[65] H. Zhao, Z. Li, H. Wei, J. Shi, and Y. Huang, "SeqFuzzer: An industrial protocol fuzzing framework from a deep learning perspective," in *Proc. 12th IEEE Conf. Softw. Test., Validation Verification (ICST)*, Apr. 2019, pp. 59–67.
[66] M. Niedermaier, F. Fischer, and A. Von Bodisco, "PropFuzz—An IT-security fuzzing framework for proprietary ICS protocols," in *Proc. Int. Conf. Appl. Electron. (AE)*, Sep. 2017, pp. 1–4.
[67] H. Huang, H. Deng, J. Chen, and X. Ye, "Study on fuzzing test strategy based on improved differential evolution algorithm," in *Proc. Int. Comput., Signals Syst. Conf. (ICOMSSC)*, Sep. 2018, pp. 655–659.
[68] J. L. Rrushi, "SCADA protocol vulnerabilities," in *Critical Infrastructure Protection*. Springer, 2012, pp. 150–176.
[69] A. Agostin. *Defense-in-Depth or How to Secure Industrial Control System Critical Infrastructure*. [Online]. Available: https://www.mtl-inst.com/images/uploads/EATON_WP_Defense_in_Depth.pdf
[70] B. Green, M. Krotofil, and A. Abbasi, "On the significance of process comprehension for conducting targeted ICS attacks," in *Proc. Workshop Cyber-Phys. Syst. Secur. PrivaCy - CPS*, 2017, pp. 57–67.
[71] wolti sil berliosrobot. *Freemodbus*. [Online]. Available: https://sourceforge.net/projects/freemodbus.berlios/files/.
[72] GeraldCombs and other contributors. *Wireshark: A Free Open Source Network Protocol Detection and Analysis Program*. [Online]. Available: https://www.wireshark.org/
[73] F. Schuster, F. M. Kopp, A. Paul, and H. Konig, "Attack and fault detection in process control communication using unsupervised machine learning," in *Proc. IEEE 16th Int. Conf. Ind. Informat. (INDIN)*, Jul. 2018, pp. 433–438.
[74] J. M. Beaver, R. C. Borges-Hink, and M. A. Buckner, "An evaluation of machine learning methods to detect malicious SCADA communications," in *Proc. 12th Int. Conf. Mach. Learn. Appl.*, Dec. 2013, pp. 54–59.
[75] S. D. D. Anton, S. Sinha, and H. Dieter Schotten, "Anomaly-based intrusion detection in industrial data with SVM and random forests," in *Proc. Int. Conf. Softw., Telecommun. Comput. Netw. (SoftCOM)*, Sep. 2019, pp. 1–6.
[76] S. D. Duque Anton, A. Pia Lohfink, C. Garth, and H. Dieter Schotten, "Security in process: Detecting attacks in industrial process data," 2019, *arXiv:1909.03730*. [Online]. Available: http://arxiv.org/abs/1909.03730
[77] G. Bernieri, M. Conti, and F. Turrin, "Evaluation of machine learning algorithms for anomaly detection in industrial networks," in *Proc. IEEE Int. Symp. Meas. Netw. (M&N)*, Jul. 2019, pp. 1–6.
[78] S. D. Anton, S. Kanoor, D. Fraunholz, and H. D. Schotten, "Evaluation of machine learning-based anomaly detection algorithms on an industrial Modbus/TCP data set," in *Proc. 13th Int. Conf. Availability, Rel. Secur. (ARES)*, 2018, p. 41.
[79] M. Zolanvari, M. A. Teixeira, L. Gupta, K. M. Khan, and R. Jain, "Machine learning-based network vulnerability analysis of industrial Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6822–6834, Aug. 2019.
[80] A. N. Sokolov, I. A. Pyatnitsky, and S. K. Alabugin, "Research of classical machine learning methods and deep learning models effectiveness in detecting anomalies of industrial control system," in *Proc. Global Smart Ind. Conf. (GloSIC)*, Nov. 2018, pp. 1–6.
[81] F. Schuster, A. Paul, R. Rietz, and H. Koenig, "Potentials of using one-class SVM for detecting protocol-specific anomalies in industrial networks," in *Proc. IEEE Symp. Ser. Comput. Intell.*, Dec. 2015, pp. 83–90.
[82] Q. Wang, H. Chen, Y. Li, and B. Vucetic, "Recent advances in machine learning-based anomaly detection for industrial control networks," in *Proc. 1st Int. Conf. Ind. Artif. Intell. (IAI)*, Jul. 2019, pp. 1–6.
[83] M. Mantere, M. Sailio, and S. Noponen, "Feature selection for machine learning based anomaly detection in industrial control system networks," in *Proc. IEEE Int. Conf. Green Comput. Commun.*, Nov. 2012, pp. 771–774.
[84] A. Cook, R. Smith, L. Maglaras, and H. Janicke, "Measuring the risk of cyber attack in industrial control systems," in *Proc. BCS eWiC*, 2016, pp. 103–113.
[85] M. Naghavi, P. Libby, E. Falk, S. W. Casscells, S. Litovsky, J. Rumberger, C. Stefanadis, P. Moreno, G. Pasterkamp, and Z. Fayad, *From vulnerable plaque to vulnerable patient: A call for new definitions risk assessment strategies: Part II*. 2003.
[86] W. S. Lee, D. L. Grosh, F. A. Tillman, and C. H. Lie, "Fault tree analysis, methods, and applications—A review," *IEEE Trans. Rel.*, vol. R-34, no. 3, pp. 194–203, Aug. 1985.
[87] S. Fine, Y. Singer, and N. Tishby, "The hierarchical hidden Markov model: Analysis and applications," *Mach. Learn.*, vol. 32, no. 1, pp. 41–62, 1998.
[88] M. J. Beal, Z. Ghahramani, and C. E. Rasmussen, "The infinite hidden Markov model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 577–584.
[89] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Mach. Learn.*, vol. 29, no. 2, pp. 131–163, Nov. 1997.
[90] B. Schneier, "Attack trees," *Dr. Dobb's J.*, vol. 24, no. 12, pp. 21–29, 1999.
[91] S. Mauw and M. Oostdijk, "Foundations of attack trees," in *Proc. Int. Conf. Inf. Secur. Cryptol.* Springer, 2005, pp. 186–198.
[92] E. J. Byres, M. Franz, and D. Miller, "The use of attack trees in assessing vulnerabilities in scada systems," in *Proc. Int. Infrastruct. Survivability Workshop*, 2004, pp. 3–10.

JIAPING MEN received the B.S. degree from the North China University of Science and Technology, China, in 1999, and the M.S. degree from Sichuan University, China, in 2010. He is currently pursuing the Ph.D. degree with the School of Computer and Information Technology, Beijing Jiaotong University, China. His main research interest includes security and privacy in cloud computing.

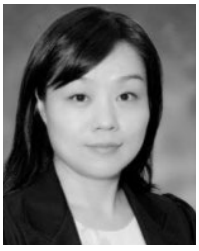
ZHUO LV is currently an Information Security Engineer with the State Grid Henan Electric Power Research Institute and the Information Division, State Grid Henan Electric Power Research Institute. He has been committed to the research of network security of electric power industry control systems for a long time, mainly responsible for the vulnerability excavation and security reinforcement of electric power industrial control systems.

XIAOJUN ZHOU received the Ph.D. degree in cyberspace security from the University of Chinese Academy of Sciences, in 2018. He is currently responsible for or participated in several industrial security projects, including a series of major scientific research projects such as the National Natural Science Foundation Projects, the National Key Research and Development Plan, and the Key Research and Development Projects of the Chinese Academy of Sciences. His research interests include the industrial control protocol security analysis, industrial Internet security, and the Internet of Things security.

ZHEN HAN received the Ph.D. degree from the China Academy of Engineering Physics, in 1991. He is currently a Professor with the School of Computer and Information Technology, Beijing Jiaotong University. He has authored or coauthored over 100 papers in various journals and international conferences. His main research interests include information security architecture and trusted computing.



HEQUN XIAN received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, in 2009. He was a Visiting Scholar with the College of Information Science and Technology, The Pennsylvania State University. His research interests include cryptography, cloud computing security, and network security.



YA-NAN SONG is currently an Associate Professor with the School of Business, Macau University of Science and Technology, and the Coordinator of International Business Major. Her research has embraced international economics and relations, with a special emphasis on Sino-Lusophone countries economic and trade cooperations. She has actively organized international symposiums on Sino-Brazil Relations and BRICS Institutionalization and planned and managed Business Investment Environments in Lusophone Countries series of lectures and projects.