Peer-reviewed version available at Electronics 2019, 8, 1545; doi:10.3390/electronics8121545

Article

A Two-Stage Intrusion Detection System for Industrial Control Networks based on EtherNet/IP

Wenbin Yu [1], Yiyin Wang [1] and Lei Song [1,*]

[1] Department of Automation, Shanghai Jiao Tong University, Shanghai, 200240 China;
* Correspondence: Email: songlei_24@sjtu.edu.cn; Tel: +86-21-34204022

Abstract: Standard Ethernet (IEEE 802.3 and the TCP/IP protocol suite) is gradually applied in industrial control system (ICS) with the development of information technology. It breaks the natural isolation of ICS, but contains no security mechanisms. An improved intrusion detection system (IDS), which is strongly correlated to specific industrial scenarios, is necessary for modern ICS. On one hand, this paper outlines three kinds of attack models, including infiltration attacks, creative forging attacks and false data injection attacks. On the other hand, a two-stage IDS is proposed, which contains a traffic prediction model and an anomaly detection model. The traffic prediction model, which is based on the autoregressive integrated moving average (ARIMA), can forecast the traffic of the ICS network in short term and detect infiltration attacks precisely according to the abnormal changes in traffic patterns. Furthermore, the anomaly detection model, using a one-class support vector machine (OCSVM), is able to detect malicious control instructions by analyzing the key field in EtherNet/IP packets. The confusion matrix is selected to testify the effectiveness of the proposed method and two other innovative IDSs are used for comparison. The experiment results show that the proposed two-stage IDS in this paper has an outstanding performance in detecting infiltration attacks, forging attacks and false data injection attacks compared with other IDSs.

Keywords: Intrusion Detection; EtherNet/IP; Industrial Control Networks

1. Introduction

Industrial control system (ICS) is composed of various automatic control components and real-time data acquisition components together. The main purpose of the ICS is for monitor and control of industrial manufacturing to ensure the normal operation of industrial equipments. The core components of the ICS include the supervisory control and data acquisition system (SCADA) [1,2], distributed control system (DCS), programmable logic controller (PLC), remote terminal unit (RTU), human machine interface (HMI) and a variety of communication interface technologies [3–5]. The ICS has been widely applied in energy industry, transportation, metallurgy, electric power system, etc.

In the traditional ICS, experienced engineers mainly focus on the physical safety of the production and ignore the information security because of the natural isolation of industrial networks, which makes it impossible for malicious hackers to interact with the traditional ICS [6–8]. With the rapid development of information technology (IT), standard Ethernet (IEEE 802.3 and the TCP/IP protocol suite) are gradually implemented in the ICS communication interface. **As a consequence, many automation companies designed standardized industrial field bus and Ethernet. The Schneider electric also released Modbus/TCP [9] to replace the previous Modbus/RTU [10], which used the original RS-485 as a communication interface. At the same time, Profinet [11], which achieved Profibus over industrial Ethernet, was defined by PROFINET International. The Rockwell automation proposed the EtherNet/IP [12], which supports data communications over industrial Ethernet, which enlarger the bandwidth of the communication. The EtherNet/IP was first introduced in 2001 and now is the most developed, mature and complete industrial Ethernet solution available for manufacturing automation, with rapid growth as users are eager to take**

advantages of the open technologies and Internet. EtherNet/IP implements the common industrial protocol (CIP) over standard IEEE 802.3 and the TCP/IP protocol suite.

Implementing standard Ethernet in the modern ICS improves the interoperability of the ICS and greatly reduces the cost of application developments. However, it also breaks the natural isolation of industrial networks. The modern ICS are facing more advanced threats from the Internet outside the factory. However, the original ICS security mechanisms, such as the industrial firewall and white list, can not handle with these threats effectively enough. On one hand, industrial firewall can not dissect industrial communication protocols (e.g. EtherNet/IP), which makes it impossible to inspect the application layer payloads in packets or automatically generate proper filter rules according to the specific industrial scenarios [6]. On the other hand, the white list can only function as an access control list and it is easy to forge as a result of many brilliant penetration testing tools, such as the Metasploit [13,14].

As a second line of defense, the intrusion detection system (IDS) is an effective approach to detect malicious intruders, who are trying to disrupt the ICS networks from the Internet. By analyzing the information collected from the key points in the network, the IDS can find out whether there is a violation of the security policies and decrease the probability of attack occurrence. According to the analysis and inspection of the problems, the IDS takes appropriate countermeasures, such as raising an alarm or blocking the suspicious connections [3].

Without a doubt, there has been also many innovative works in designing the IDS for industrial networks. [15] designed a telemetry based IDS by measuring the statistical data about client server sessions from the traffic flow. Although it was practical to detect anomaly in the ICS networks, it had a strong precondition that the existence of the time delays was introduced by spoofing. Apart from this, the data related to the network protocols, which was more valuable, were not utilized to design the IDS. [16] constructed an anomaly-based IDS according to the normal behaviors of function control and process data. The behavior extraction algorithm was attractive to researchers because it considered the information entropy of function code used by the Modbus/TCP protocol. However, the IDS used function code sequence in the time interval as input. If a packet was fabricated, which contained the same function code in the same time interval but at a wrong time point, it would be impossible to detect the fake packets, which could be a serious threat to the ICS. [17] built a model for normal system behavior to distinguish the normal and abnormal system operations. **All these methods did not consider the characteristics of the data traffic and the normality modeling. In order to overcome the shortages of previous works, it is required to construct an intrusion detection system which could reflect the behavior characteristics in the ICS networks, strongly correlated to the ICS protocols and be able to cope with vulnerabilities. Furthermore, it should have a satisfactory overall accuracy and false alarm rate.**

Above all, it is better to think like a hacker before stopping a hacker. This paper firstly considers the infiltration attacks, forging attacks and false injection attacks. In addition to this, an EtherNet/IP structure is fabricated explicit messaging that use the TCP as the transmission protocol. To prevent the attacks mentioned above, this paper designs a two-stage intrusion detection system for the ICS based on the EtherNet/IP. The two-stage IDS has ability to dissect the EtherNet/IP protocol and mainly contains a traffic prediction model and an anomaly detection model. The traffic prediction model based on the autoregressive integrated moving average (ARIMA) can protect the ICS networks from infiltration attacks. The anomaly detection model based on one-class support vector machine (OCSVM) is able to detect the elegant fabricated EtherNet/IP packets and protect against the.forging attacks and false injection attacks. Compared with other creative IDSs [15] [16], the proposed method performs a satisfactory results in terms of overall accuracy and false alarm rate.

The rest of this paper is organized as follows: Section 2 introduces related works. Section 3 describes the simulated industrial scenario and make a brief introduction about the EtherNet/IP protocol. Section 4 outlines the attack models, especially the fabricating malicious EtherNet/IP packets. Section 5 elaborates on the two-stage intrusion detection system, which consists of a traffic prediction

model and an anomaly detection model. Section 6 is the simulations and analysis. Finally, Section 7 makes a conclusion of this paper.

2. Related Works

As a hot research topic, many researchers are focusing on the two-stage intrusion detection approaches. [18] studied a classical chemical process and evergo all the attack vectors. A data-driven approach to detect anomalies was designed for early indicators of malicious activity. The model of the attack process was built to profile some kinds of default disturbances. [19] proposed a machine learning based approach, which could reduce the amount of manual customization required for different ICS networks. Also, a series of features for the machine learning input is selected and the structure was used for real industrial process control network. [20] designed an SCADA based IDS, which could deal with the Denial of Service Attacks. The network was investigated to resist response injection and denial of service attacks. A periodicity characteristics analyzing algorithm was designed for SCADA networks and this feature was used for intrusion detection. Similarly, these papers all considered to detect the intrusion by modeling the normal operation features, which is also the main idea of this paper. Differently, this paper focuses on the traffic of the ICS network based on the control and data stream. Also, the application scenarios in this paper are basic data transmission for the industry network.

After the normal transmission traffic is modeled, a classifier is required to be designed to judge whether the receiving data are normal or attack data. [21] compared various machine learning algorithms to reduce the False Alarm Rate and maintain high accuracy for industrial intrusion detection. [22] introduced a machine learning based classifier to reduce the false alarm and guarantee the precision for intrusion detection in industry area. [23] focused on the IDS and compared different techniques for Industrial Internet of Things including machine learning and non-machine learning methods. The machine learning structure is a widely used method for industrial intrusion detection. However, as is known to all, the training for the structure requires large quantity of labeled data. In this paper, an SVM based structure is designed and an optimization problem is formulated to build the classifier, which can make full use of the labeled data.

3. Industrial Scenario

In order to make the proposed approach more effective and practical in the real ICS, an ICS demo platform is established based on the EtherNet/IP. As shown in Fig. 1, the platform consists of three layers: process monitor layer, process control layer and plant-floor layer. The process monitor layer contains the operation station, where engineers can monitor the entire manufacturing procedure and modify the program or parameters in the PLC. This layer also contains the malicious attacker and the intrusion detection system (IDS). The process control layer in the middle constitutes a set of devices that serve the productions, including the PLC, industrial router and variable frequency drive (VFD). The PLC is the Allen-Bradley Micro 850 series and the VFD is PowerFlex 525 from Rockwell. The plant-floor layer consists of a whole Auto-guided Vehicle (AGV) system, which includes an optical-electricity encoder and three-phase asynchronous motor encoders.

In this ICS, the AGV system is regarded as the controlled object and the controller is AB Micro 850 PLC. The feedback from the optical-electricity encoder are used to determine the current position of the encoder and the speed of the motors. The motions of the AGV system is controlled by the motors, which are driven by the PowerFlex 525 VFD. The entire process can be summarized as follows: according to the feedback from the optical-electricity encoder, the PLC makes decisions and sends EtherNet/IP control packets to the VFD and the VFD adjusts the motors speed and position by running the motors. This is a precision linear motion mechanism that converts the motor rotation into linear motion and it is widely used in various linear motion applications.

The utilized communication protocol is EtherNet/IP. The physical layer and data link layer are based on the Ethernet, while the transport layer and network layer are based on the TCP/IP protocol

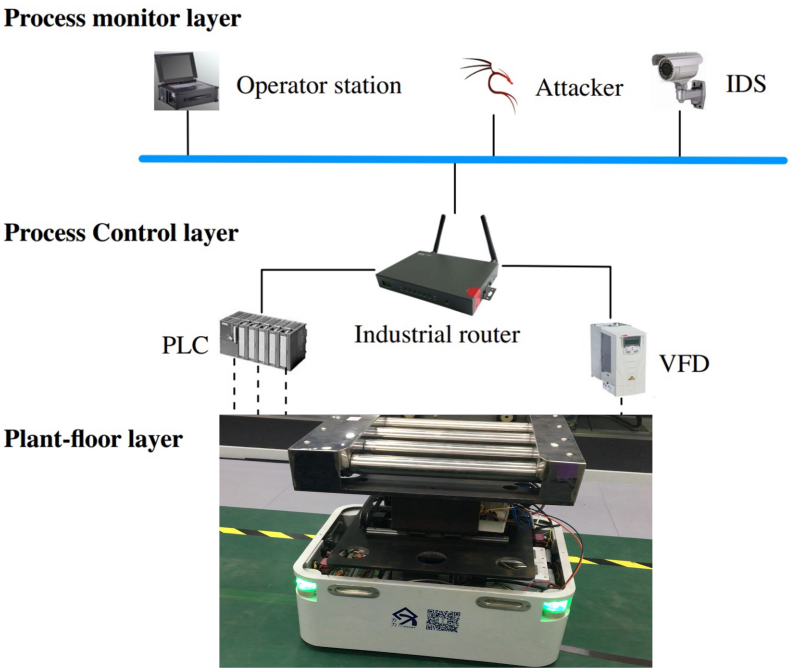


Figure 1. Simulated ICS based on EtherNet/IP

suite including transmission control protocol (TCP), user datagram protocol (UDP), Internet protocol (IP) and address resolution protocol (ARP), etc. The CIP is used as application layer and it defines two primary types of communications: implicit and explicit messaging. The implicit messaging is often used to transfer real-time control data from a remote I/O device with UDP, while explicit messaging, which is mainly discussed in this paper, is utilized with the TCP for request/reply transactions [24].

4. Attack Models

In this section, some kinds of infiltration attacks will be modeled. Besides that, it is creatively proposed that a technique to fabricate an EtherNet/IP packet containing explicit messaging with the help of scapy, which is a powerful interactive packet manipulation program. The attack platform is based on the Kali Linux, which is an advanced penetration testing Linux distribution. All the EtherNet/IP packets are parsed out by using the python scripts. However, this paper mainly focuses on the cyber-attacks and safety protection after entering the network.

4.1. Infiltration Attacks

Port scanning is usually used to identify some services and systems in the traditional network. By sending a TCP SYN packet to establish a connection at each port, it can be recognized that the port has been opened according to the response. Through the open situation of the port, it is possible to understand what services and operating system are running. As utilized port scanning in target ICS system, the results show that the TCP port number (e.g. port 44818) is open, which indicates that the EtherNet/IP service is running on this ICS. However, the information collected by port scanning is incomplete and device enumerate is needed to identify the ICS devices.

Device enumerate is used to identify the device information in the target ICS and it is achieved by sending an EtherNet/IP packet to the remote device that has some TCP port number open. The packet is a request, using EtherNet/IP list identity command, whose function code is, e.g., 0x63. Once a response is received, the information will be parsed out, including the vendor ID, device type, product name, device IP, etc. As it is achieved device enumerate in the ICS, the PLC, VFD and the corresponding address can be identified.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 12 December 2019 doi:10.20944/preprints201912.0174.v1

Peer-reviewed version available at Electronics 2019, 8, 1545; doi:10.3390/electronics8121545

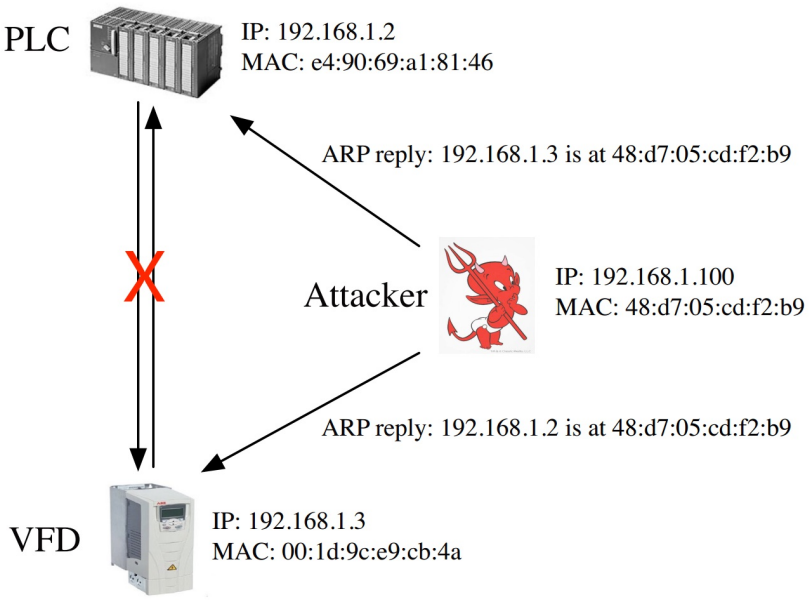


Figure 2. ARP spoof hijack EtherNet/IP session

According to the results of the device enumerate, the ARP spoof can be implemented to hijack the EtherNet/IP session between the PLC and VFD. The ARP is used to convert an IP address (network layer address) to MAC address (link layer address). By sending malicious ARP reply, attackers can disguise as a VFD in front of a PLC as shown in Fig. 2 and spoof the VFD in the same way. With the ARP spoof, attackers can hijack the EtherNet/IP session and monitor the data flow between the PLC and VFD. The communication packets between the PLC and VFD are based on explicit messaging in the EtherNet/IP. It mainly includes two parts: periodic maintenance and control instructions. **The periodic maintenance packets refer to the read and write messaging for device parameters, which use class code (e.g. 0x93) and the whole period including request. The control instructions contain rotate clockwise with function code (e.g. 0x2a) , rotate anticlockwise with, e.g. 0x29, stall clockwise with, e.g. 0x1a and stall anticlockwise with, e.g. 0x19. The specific rotate speed is appended after function code.**

In order to avoid raising an alarm and disturb the production process, it is better to select the forwarding periodic maintenance packets and discard the control instruction packets at the same time. Selective Forwarding maintenance messages can ensure that the interface of the VFD has EtherNet/IP traffic and it will not report errors. Dropping the control messages will prevent the PLC from controlling the VFD and destroy the production process.

4.2. Forging Attack - Fabricate EtherNet/IP Explicit Message

As mentioned in Section 3, the EtherNet/IP defines two kinds of communications: implicit and explicit messaging. The communication between the PLC and VFD in the ICS is based on explicit messaging because it used TCP as the transport layer protocol. The TCP provides reliable, ordered and error-checked delivery of a stream of octets between the PLC and VFD and it is impossible to inject fake data [25–27].

However, this paper proposes a technique to inject a fabricated EtherNet/IP explicit messaging. Algorithm 1 shows the principle of the proposed mechanism. The algorithm is realized by writing a tailored python script and the third party library used here is scapy, which is an elegant packet manipulation tool for networks. The inputs contain IP addresses of the PLC and VFD and specific control instruction, which are acquired by infiltration attacks mentioned above.

The forging attack consists of three steps. Firstly, capturing any EtherNet/IP packets to get the session handle in the transaction between the PLC and VFD, where session handle can be obtained by

Algorithm 1 Fabricate EtherNet/IP packet using Scapy

function FORGE ENIP PACKET(*plc ip*, *vfd ip*, *control*)
    enip_pkt ← sniff(filter: TCP port 44818)
    session_handle ← *enip_pkt*[session]
    ack_pkt ← sniff(filter: ip src *plc_ip* and dst port 44818 and length 64)
    seq← *ack_pkt*[TCP][seq]
    ack← *ack_pkt*[TCP][ack]
    sport← *ack_pkt*[TCP][sport]
    ip_id← *ack_pkt*[IP][id] + 1
    enip_data← unhexlify(*control*, *session handle*)
    forged_pkt← IP(src: *plc_ip*, dst: *vfd_ip*, id: *ip_id*) + TCP(sport: *sport*, dport: 44818, seq: *seq*, ack: *ack*, flags: PSH and ACK) + *enip_data*
    send(*forged_pkt*)
end function

dissecting the session handle field in encapsulation header of the EtherNet/IP. Secondly, capturing the TCP ACK packet sent by the PLC to obtain some key fields (seq, ack, TCP source port, IP identification). Thirdly, forge and send the EtherNet/IP packet according to the information previously obtained. The forged packet contains certain control instructions (e.g. 0x2a00dc05), which indicates rotating clockwise at a specific rotation speed.

4.3. False Data Injection Attack

As mentioned in Section 3, the EtherNet/IP based network structure will transmit several kinds of data including an optical-electricity encoder and three-phase asynchronous motor encoders. The false data injection attacks try to damage the sampled data and change the control instructions by inject false data. The attack vector is defined as $a = [a_1, a_2, ..., a_m]^T$ and the sampled data will be changed as:

$$s^a = s + a = Hx + e + a \tag{1}$$

When the false data injection attack is operating, the attack vector a is non-zero and the state error vector is c and the sampled data value will be changed to $s + a$. The proposed attack model is focusing on the data collecting and the motion control instructions. When the attack vector satisfies $a = Hc$, the attack model can bypass the classical attack detection approaches [28]. The attack model is defined as:

$$\begin{aligned} r_a =& ||s^a - H(\hat{x} + c)|| \\ =& ||s - H\hat{x} + a - Hc|| \\ \leq& ||s - H\hat{x}|| + ||a - Hc|| \\ =& r + \tau_a, \end{aligned} \tag{2}$$

in which r_a indicates the error sampled after the false data injection attack works. τ_a indicates the error change caused by the attack vector. When $\tau_a = 0$, the attack can not be detected because no change will be caused on the real sample data value. Also, the attack strategy can perform attack at the data collecting control by change the sensor ID or even the sensor type.

5. Two-Stage IDS

The attack models proposed in Section 4 are feasible and practical because the EtherNet/IP does not define any explicit or implicit security mechanisms. In order to protect the ICS from these threats, it is proposed a two-stage IDS for the ICS networks based on the EtherNet/IP. The two-stage IDS is located in process monitor layer as shown in Fig. 1 and the work flow of it is shown in Fig. 3. The inputs are captured packets in one time slot, which can be modified by users. The two-stage IDS has

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 12 December 2019 doi:10.20944/preprints201912.0174.v1

Peer-reviewed version available at Electronics 2019, 8, 1545; doi:10.3390/electronics8121545

ability to capture and analyze all communication packets from the monitoring port on the industrial router. Besides that, it uses libpcap technique to capture packets and dissects all the EtherNet/IP packets by running our python scripts. In the python scripts, the EtherNet/IP packets layer is dissected by layer and obtain necessary information according to the key fields, such as control instructions occurred in Section 4.

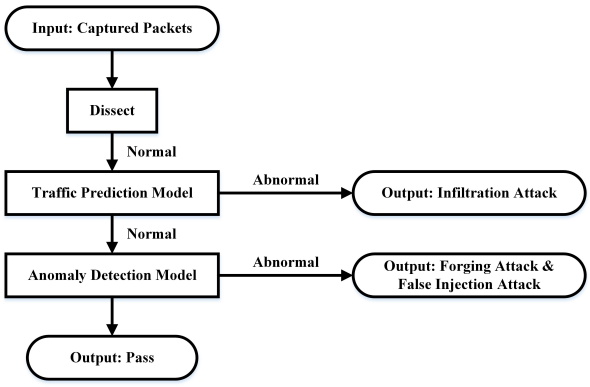


Figure 3. The work flow of two-stage IDS

The two-stage IDS mainly consists of two parts: the traffic prediction model and anomaly detection model. The traffic prediction model based on the ARIMA model is designed to estimate the number of packets in next time slot according to the captured packet flow previously. It is feasible to detect the infiltration attacks listed in section 4 because the infiltration attacks will lead to abnormal increase or decrease of the ICS network traffics at a specific time slot. If abnormal instances do not occur, the packets will deliver to anomaly detection model. That is because there is still possible for a forging attack, which does not lead to traffic changes. The anomaly detection model based on OCSVM is practical to detect the proposed forged EtherNet/IP packet because it could acquire essential control instruction by analyzing theEtherNet/IP packets in depth and detect abnormal behaviors by comparing with the normal pattern.

5.1. Traffic Prediction Model

A well designed traffic prediction model can precisely reflect the traffic characteristics of the ICS network. Since infiltration attacks lead to abnormal traffic fluctuations, it is possible to be detected with the traffic prediction model. The traffic flow data is a kind of time series and the ARIMA model is the most commonly utilized for time series prediction. The ARIMA model is used for short-term forecasting and the fundamental patterns of the time series should not be changed, which means the ICS networks should be immutable and the production process is stable.

The raw input time series of the ARIMA, which is the count of traffic packets in one time slot, is not stable and fluctuate periodically. In order to use the ARIMA model, it is required to preprocess the raw input by using logarithmic transformation and differentiating. Logarithmic transformation is mainly to reduce the vibration amplitude of the sequence, making the linear rule more obvious. Differentiating is able to make the series stable and the difference periods are the periods of the ICS. Augmented Dickey-Fuller test is used to test stationarity of the time series. Furthermore, Ljung-Box test for autocorrelation is essential to ensure that the time series is not white noise. After preprocessing and testing, a stable and non-white noise time series $\{x_t\}_{t=1}^{n}$ is obtained and it is suitable for the ARIMA model.

The prediction function of ARIMA can be depicted as:

$$\hat{x}_t = \psi_1 x_{t-1} + \psi_2 x_{t-2} + ... + \psi_p x_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + ... + \theta_q \epsilon_{t-q}, \quad (3)$$

where x_t is the stable and non-white noise time series, p and q are the order of autoregressive (AR) model and moving average (MA) model. ψ_i and θ_i are the parameters of the ARIMA model. The prediction error must be uncorrelated and obey normal distribution $\epsilon_t \sim N(0, \sigma^2)$.

The order p and q determine the accuracy of the ARIMA model and they can be estimated by calculating autocorrelation function (ACF) and partial autocorrelation function (PACF) [29].

In addition to the ACF and PACF, enumerate many ARIMA models with different orders and use some criterion can also determine the proper model, i.e. the optimal parameters p and q. The criterion contains Akaike information criterion (AIC), Bayes information criterion (BIC) and Hannan-Quinn information criterion (HQIC). While the statistical ideas of these criterion are the same, that is considering the fitting of the residuals and imposing punishments related to the number of variables at the same time. After calculating the predicted value x_t, it is recovered the traffic prediction values by the inverse difference and exponentiation. The results also need to be rounded to the the nearest integer.

5.2. Anomaly Detection Model

Anomaly detection model acts as a second line of defense after traffic prediction model. Malicious attackers may drop the original EtherNet/IP control packet and replace it with the fabricated one that contains wrong control instructions. Forging attack and false data injection attack can not be detected by traffic prediction model because of it has little impact on traffic flow. Therefore, it is necessary to establish an anomaly detection model for the forging attack and false data injection attack.

The anomaly detection model firstly filters out the EtherNet/IP control packets according to the field of service. The service name of the control packets is set attribute single, whose code is e.g. 0x10. After obtaining the control packets, specific control instructions should be extracted from the packets. The control instructions have 4 features, i.e. relative time, action, direction and speed. The relative time refers to the packet time stamp relative to the control period, which is a control cycle for the application. The action refers to rotate, whose value is e.g. 0x2 or stall (0x1) and the direction refers to clockwise (0xa) or counterclockwise (0x9). The speed simply refers to the rotation speed. After feature selection, the feature samples (control instructions) are obtained $\{\mathbf{x}_i\}_{i=1}^N$ and each sample $\mathbf{x}_i$ with the 4 features.

In order to detect the forged packets with wrong control instructions, an OCSVM is constructed with the collected samples which are all normal data. The OCSVM is a modified algorithm based on the SVM and has been widely used for one-class classification problem, such as the anomaly detection. According to [30–32], the OCSVM firstly maps a sample $\mathbf{x}_i$ from input space to the feature space F using Kernel function. Feature space has a higher dimensions and the separation may be easier in feature space. Secondly, the OCSVM considers origin as abnormal and training samples as normal, and constructs an optimal hyperplane between normal and abnormal by maximize the margin.

The OCSVM mainly resolves the following quadratic programming optimization problem:

$$\begin{aligned} \min_{w \in F} \quad & \frac{1}{2}||\mathbf{w}||^2 + \frac{1}{vn}\sum_{i=1}^{n}\xi_i - \rho \\ s.t. \quad & \langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle \geq \rho - \xi_i, \xi_i > 0, i = 1...n, \end{aligned} \tag{4}$$

in which $\mathbf{w}$ is the normal vector in feature space; n is the number of training samples;; ξ_i is the slack variable to handle outliers and $\phi(\cdot)$ is the mapping function mentioned above. ρ is the compensation parameter and v defines the upper bound on the fraction of training errors and a lower bound of the fraction of support vectors.

By utilizing Lagrangian method, the dual formulation and transform the original problem are obtained to calculating Lagrangian operator $\{\alpha_i\}_{i=1}^n$. The Lagrangian operator can be resolved by sequential minimal optimization (SMO) [33–35]. Finally, the decision function can be obtained using Kernel method:

$$f(\mathbf{x}) = \langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle - \rho = \sum_{i=1}^{n} \alpha_i \Phi(\mathbf{x}_i, \mathbf{x}) - \rho, \tag{5}$$

where $\Phi(\mathbf{x}_i, \mathbf{x}_j) == \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$ is the kernel function. For any new input sample $\mathbf{x}$, if $f(\mathbf{x}) \geq 0$, then $\mathbf{x}$ is labeled as normal. Otherwise, if anomaly instances are detected, it indicates that the input control instruction is different from the normal behavior and it is likely to be a fake packet. The anomaly detection model is implemented based on the OCSVM using python. The kernel function selected is Gaussian kernel as shown in the following formulation:

$$\Phi(\mathbf{x}_i, \mathbf{x}_j) = exp\left(-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{\gamma}\right). \tag{6}$$

6. Simulations

6.1. Scenarios

The two-stage IDS is tested in the ICS network based on EtherNet/IP. As mentioned in Section 3, the ICS platform is built for experiments to testify the effectiveness of the proposed approach. The platform consists of three layers: process monitor layer, process control layer and plant-floor layer. The process monitor layer contains the malicious attacker and IDS. The process control layer constitutes the PLC, industrial router and VFD. The PLC used is Allen-Bradley Micro 850 series and VFD used is PowerFlex 525 from Rockwell. Moreover, the network structure is used to transmit an AGV localization data including an optical-electricity encoder and three-phase asynchronous motor encoders. In this paper, according to the technological requirements, the time slot is selected to be 1 second and the ICS is running for 1 day including 86400 time slots.

The packets extracted from each time slot are delivered to the IDS for real-time inspection, which lasts for less than 10^{-4} second. 1000 infiltration attacks, 1000 forging attacks and 1000 false data injection attacks are randomly launched during the day. Also, the corresponding 3000 normal instances are used for the simulation. The simulation results are displayed using confusion matrix, which is as shown in Table 1. The confusion matrix is designed for a two-class classifier and is useful to evaluate the performance of the IDS [36].

Table 1. Confusion Matrix for IDS System

Actual Class	Predicted Class	
Class	Normal	Attack
Normal	True negative (TN)	False Positive (FP)
Attack	False Negative (FN)	True positive (TP)

6.2. Simulation Results

6.2.1. Metrics

After calculating the predicted value x_t, the traffic prediction values are recover by inverse difference and exponentiation. The results also need to be rounded to the nearest integer. The traffic prediction model in real ICS network is implemented and the results of the rolling prediction are depicted in Fig 4. Root mean square error (RMSE), which is calculated by:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n} (x_t - \hat{x}_t)^2}{n}}, \tag{7}$$

which is used to determine the threshold between normal and abnormal status. At time slot t, if the difference $d_t = x_t - \hat{x}^t$ is larger than $1.5 * RMSE$, the ICS network is suspected to be infiltration attacked and the two-stage IDS will notify the engineer to check the traffic log. If no anomaly detected, the packets will deliver to anomaly detection model for the next step inspection. The kernel coefficient is set to be 0.1 according to validation. The training samples (control instructions) are extracted from the ICS network in normal operation and the smallest training error is to be guaranteed.

To evaluate the performance of the proposed method, four metrics are selected, the Overall Accuracy decision rates, False Positive Rate, False Negative Rate and Precision Rate [36,37].

The Overall Accuracy is defined as:

$$Overall Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{8}$$

which demonstrates the accuracy of the behavior of the attack detection. Literally, the overall accuracy denotes all the correct classifications against all the classifications.

The False Positive Rate (FPR) is defined by:

$$FPR = \frac{FP}{TN + FP}, \tag{9}$$

which describes that the rate of the wrong predictions for the normal instances. The FPR can also be denoted by Fallout Rate [37].

Conversely, the False Negative Rate (FNR) is defined as:

$$FNR = \frac{FN}{TP + FN}, \tag{10}$$

which describes that the rate of the wrong predictions for the normal instances. The FNR can also be denoted by Miss Rate [37].

Furthermore, the Precision Rate (PR) is defined as:

$$PR = \frac{TP}{TP + FP}, \tag{11}$$

which describes that the precision of the prediction when an attack prediction is made [37].

All the above metrics are used for simulations comparing with other IDS methods.

6.2.2. Performance of the Traffic Prediction Model

The order p and q determine the accuracy of the ARIMA model and they can be estimated by calculating autocorrelation function (ACF) and the partial autocorrelation function (PACF) [29]. According to the results, both the ACF and PACF have trailing characteristics, and they both have obvious first order correlations. So set p = 1 and q = 1.

It is simulated that the ARIMA model using python and fit model by exact maximum likelihood via Kalman filter. After comparing the models by criterion, the selected model is ARIMA(3,1,1) where p = 3, q = 1 and d = 1 which indicates first difference.The given order p and q are different from the order observed by the ACF and PACF. The results of the observation are partly influenced by the real operation data. Furthermore, the ARIMA model need to be updated periodically and it is more convenience to select proper model by using the criterion than observing ACF and PACF. The criterion of the selected model are small enough and satisfactory.

$$AIC = -867.42, BIC = -887.15, HQIC = -875.41$$

The final parameters are as follows:

$$\psi_1 = -0.8561, \psi_2 = 0.6794, \psi_3 = -0.446, \theta_1 = -0.7998$$

And then, the prediction function can be calculated by the following equation.

$$\hat{x}_t = -0.8561x_{t-1} + 0.6794x_{t-2} - 0.446x_{t-3} + \epsilon_t - 0.7998\epsilon_{t-1}, \quad (12)$$

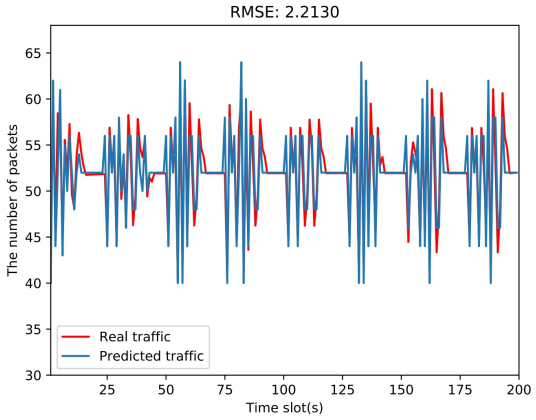


Figure 4. The result of traffic prediction model based on ARIMA

6.2.3. Performance of the Anomaly Detection Model

In this section, the performance of the OCSVM is tested. As the proposed OCSVM approach is the kind of classification algorithm, the other two machine learning based methods are selected as the comparisons, which are called Semi-supervised machine learning method [22] and the Boosting based machine learning method [28].

As the size of the training data seriously affect the performance of each classification approach. In order to simulate the methods equally, three data sizes are selected to be labeled training data, 877 (380 minutes), 3323 (24 hours) and 6646 (48 hours).

The simulation results are as shown in Fig. 5. It is depicted in Fig. 5 that almost all the algorithms are improving as the training data set size increases. When the data size is large, the Overall Accuracy and Precision Rate of all the methods can obtain an acceptable performance, both the Overall Accuracy and Precision Rate are more than 85%, which is shown in Fig. 5(c). However, when the data size is small, the proposed method will perform better than the others, which is shown in Fig. 5(a). Furthermore, when the data size increases for about 6 times, the FPR of the two machine learning methods drops from the highest more than 20% to below 5%. Also, the FNR drops from more than 35% to 6%. It means that as the data size increases, the machine learning based will perform better and the least data size will be selected as 6646. Comparatively, the proposed method in this paper perform much better when the data size is small. As the data size increases, the performance even becomes less effective, which may because as the training data size is enlarged, the classification model is overfitted.

6.2.4. Performance of the Proposed Two-Stage IDS

A comparison is made among our two-stage IDS, telemetry-based IDS [15] and double behavior characteristics IDS [16]. The simulation results of the confusion matrices the are shown in Table 2.

The Table 2 illustrates that the performance of the proposed method is better than the other two algorithms. For the infiltration attack, the proposed method will obtain the highest Precision Rate (higher value of TP) and the Double behavior characteristics IDS will have the highest TN, which will reflect to the Overall Accuracy but the low TP will also undermine the Overall Accuracy. For both forging and false data injection attacks, the proposed two-stage IDS will result in best

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 12 December 2019 doi:10.20944/preprints201912.0174.v1

Peer-reviewed version available at Electronics 2019, 8, 1545; doi:10.3390/electronics8121545

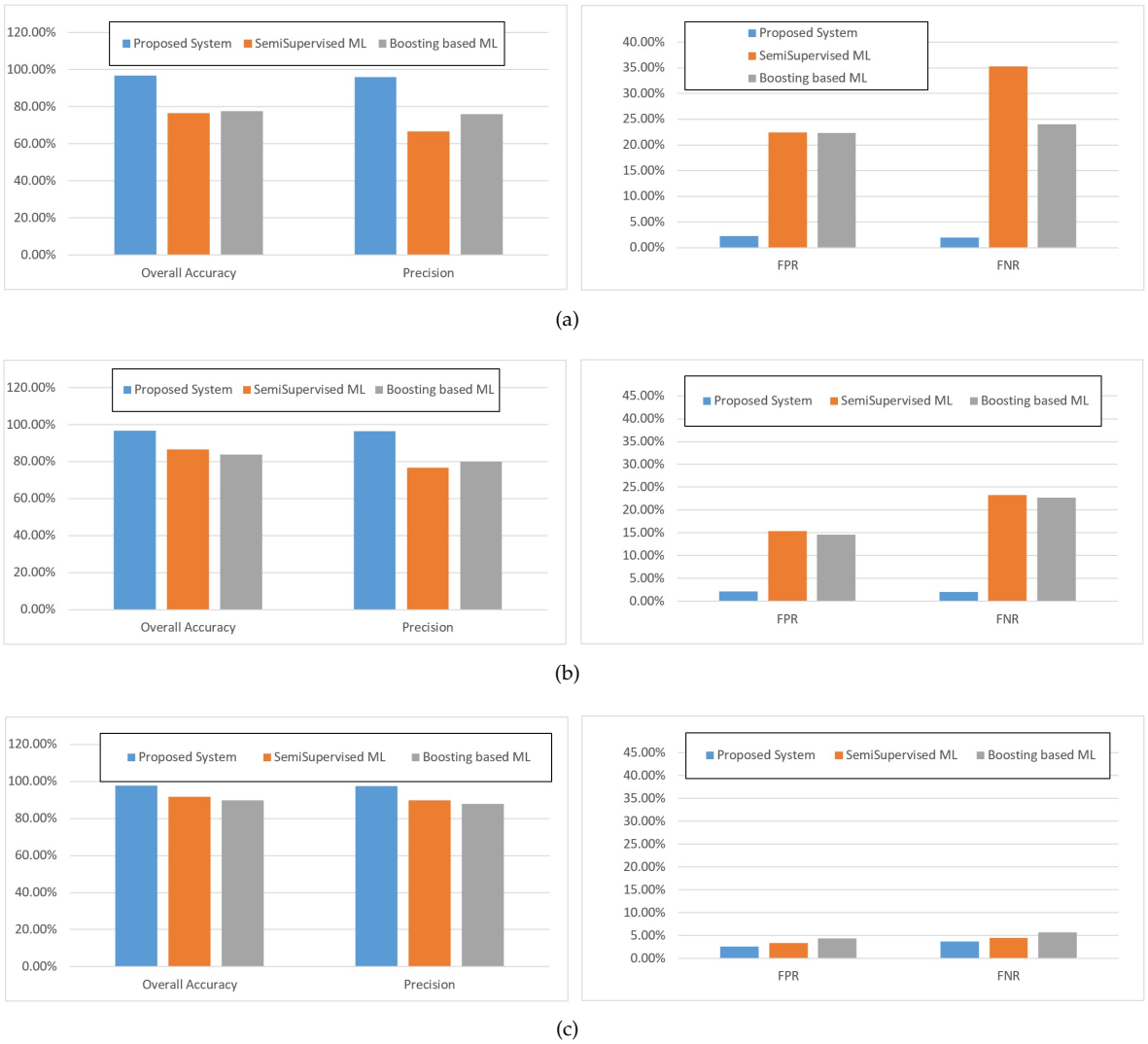


Figure 5. The classification experiment results between OCSVM and machine learning methods. (a) Train data size=877 . (b) Train data size=3323. (c) Train data size=6646.

performances. Although the defense to the false data injection attack may not be as good as the other two kinds of attacks, it still works much better than the other IDSs. The more detailed analysis is as follows and the experimental results are illustrated in Fig. 6 to Fig. 8, which is simulated under the proposed attack model mentioned in Section 4.

As shown in Fig. 6, when the simulation is carried on under infiltration attacks, the proposed Two-Stage IDS approach perform much better than the other two IDS methods, which is because the proposed method can detect the infiltration attacks by the traffic detection model. For the Overall Accuracy, the selected three methods all perform well and the accuracy of the proposed method is better than 95%. Correspondingly, the Precision Rate performance of the proposed method is about 20% better than the other two methods, which means the proposed method can predict attack instance more precisely, as shown in Fig. 6(a). Also, the double behavior characteristics IDS method can perform well in FPR metrics and not very well for FNR, which is because this method can classify more normal data into attack group, as shown in Fig. 6(b). According to the simulation results, the proposed method can detect the infiltration attacks precisely.

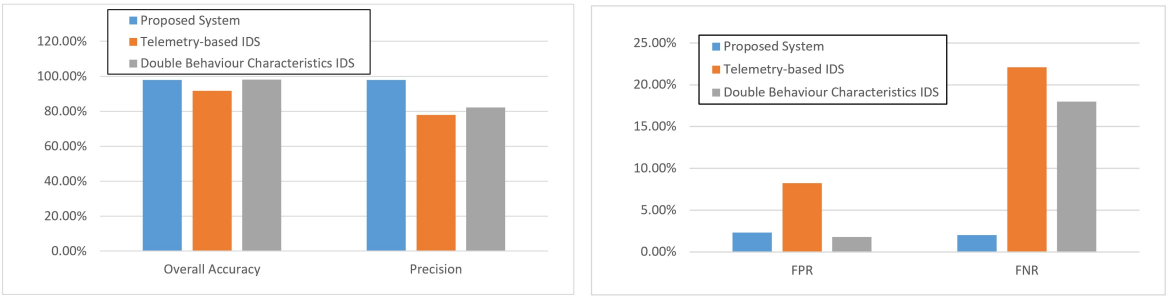
As shown in Fig. 7, when the simulation is carried on under forging attacks, the proposed Two-Stage IDS approach perform much better than the other two IDS methods, which is because the proposed method can detect the forging attacks by the traffic detection model. The selected three

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 12 December 2019 doi:10.20944/preprints201912.0174.v1

Peer-reviewed version available at Electronics 2019, 8, 1545; doi:10.3390/electronics8121545

Table 2. Confusion Matrix Results for IDS System

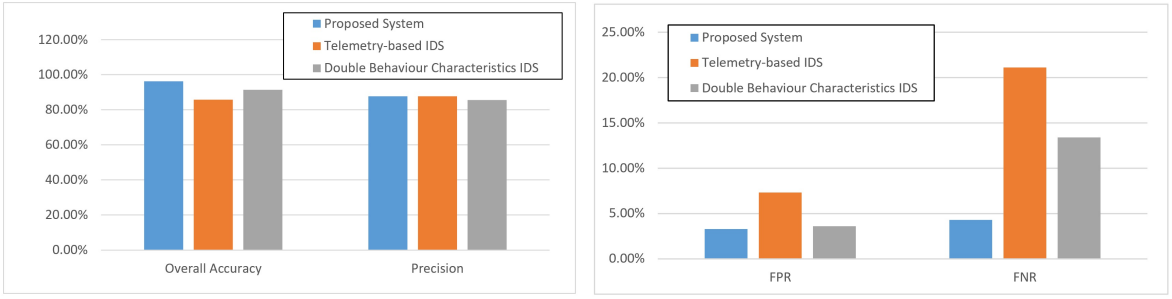
Attacks	Infiltration				Forging				False Data Injection			
Confusion Matrix Parameters	TN	TP	FN	FP	TN	TP	FN	FP	TN	TP	FN	FP
Proposed two-stage IDS	977	**980**	**20**	23	**967**	**957**	**43**	**33**	**911**	**879**	**121**	**89**
Telemetry-based IDS	918	779	221	82	927	789	211	73	768	878	122	232
Double Behavior Characteristics IDS	**982**	821	179	**18**	964	866	134	36	837	855	145	163



(a) (b)

Figure 6. Experimental results under infiltration attacks.

methods all have good performance for the Overall Accuracy. The proposed method perform as well as the double behaviour characteristics IDS around 90%, which is shown in Fig. 7(a). To be mentioned, the proposed method can maintain high Precision Rate also, because the Proposed OCSVM methods can precisely extract the features of the forging attack. On the other hand, double behavior characteristics IDS has a competitive performance in term of FPR, but the FNR is much more than that of our two-stage IDS because it is impossible to detect forging attack that is launched in a malicious time point. The proposed method has an outstanding performance in detecting the infiltration attacks and forging attack because of its models strongly related to the ICS scenario, which is shown in Fig. 7(b). The two-stage IDS processes the data collected from ICS protocol and be capable to precisely reflect the behavior characteristics in ICS network.

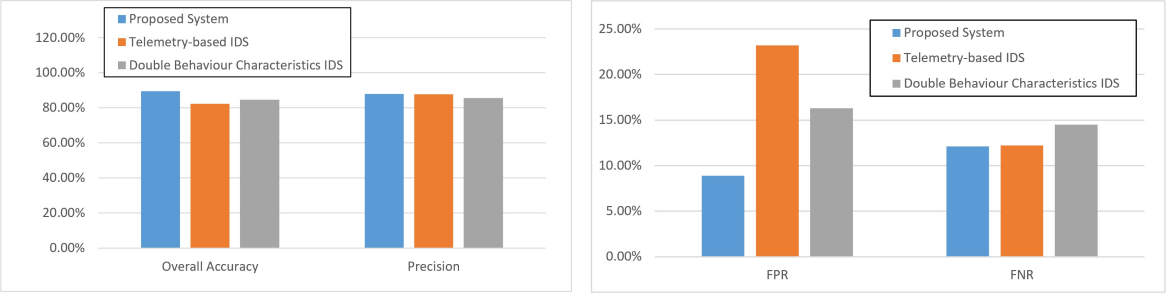


(a) (b)

Figure 7. Experimental results under forging attacks.

As shown in Fig. 8, when the simulation is carried on under false data injection attacks, it is hard to conclude that the proposed Two-Stage IDS approach perform much better than the other two IDS methods, which is because the false data injection attacks can conseal itself by changing its parameters and the proposed approach the traffic detection model and the other two methods can not precisely detect the attacks. The proposed method can maintain the Overall Accuracy and Precision Rate over 80%, as shown in Fig.8(a). Both the FPR and the FNR are a little bit higher than the former two kinds of attacks, which is shown in Fig. 8(b). To discuss more, the two-stage IDS processes the data collected from ICS protocol and be capable to precisely reflect the behavior characteristics in ICS network. If the

attack models perform differently from the normal traffic, the proposed method will perform better. Otherwise, the method will not detect all the attack data or misjudge the normal data.



(a) (b)

Figure 8. Experimental results under false data injection attacks.

7. Conclusions

This paper proposes a two-stage IDS for the ICS based on EtherNet/IP. The two-stage IDS contains a traffic prediction model and an anomaly detection model. Compared with machine learning methods, the proposed method can distinguish normal data and attack data with much less data. Also, compared with telemetry-based IDS and double behavior characteristics IDS, it offers excellent performance in detecting infiltration attacks and forging attack. To be mentioned, the performance under false data injection attack is not as good as the above two attack models. Furthermore, the proposed approach can not cope with the situation of asynchronous protocols such as the IEC 60870-5 series, which are mainly used in energy distribution networks and others because the transmission time must be known before two-stage IDS is ready to work. The future work can be done in this specific area.

The future work can be divided into two parts. On the one hand, evaluate the two-stage IDS performance in a complex ICS scenario, which contains more controllers and actuators, on the other hand, refine the two-stage IDS to defend I/O data transfers based on EtherNet/IP.

Author Contributions: Conceptualization, W.Y., and L.S.; methodology, Y.W. and W.Y.; system, W.Y.; validation, Y.W. and L.S.; experiment, Y.W.; analysis, Y.W. and W.Y.; writing—original draft preparation, Y.W. and W.Y.; writing—review and editing, L.S.

Funding: The research is sponsored by National Key Research and Development Program of China (2018YFB1308304, 2017YFB1301103), National Natural Science Foundation of China (61803261), Shanghai Natural Science Foundation of China (18ZR1421100).

Acknowledgments: The authors would like to thank and appreciate the support of all the scholars for helping us with this piece of paper and against the encountered problems.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Almalawi, A.; Tari, Z.; Fahad, A.; Khalil, I. A Framework for Improving the Accuracy of Unsupervised Intrusion Detection for SCADA Systems. 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, 2013, pp. 292–301. doi:10.1109/TrustCom.2013.40.
2. Eigner Oliver, Kreimel Philipp, T.P. Identifying S7comm Protocol Data Injection Attacks in Cyber-Physical Systems. 2018 Proceedings of the 5th International Symposium for ICS & SCADA Cyber Security Research, 2018.
3. Kargl, F.; van der Heijden, R.W.; König, H.; Valdes, A.; Dacier, M.C. Insights on the Security and Dependability of Industrial Control Systems. *IEEE Security Privacy* **2014**, *12*, 75–78. doi:10.1109/MSP.2014.120.

4. Berhe, A.B.; Kim, K.; Tizazu, G.A. Industrial control system security framework for ethiopia. 2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN), 2017, pp. 814–817. doi:10.1109/ICUFN.2017.7993912.
5. Paridari, K.; O'Mahony, N.; El-Din Mady, A.; Chabukswar, R.; Boubekeur, M.; Sandberg, H. A Framework for Attack-Resilient Industrial Control Systems: Attack Detection and Controller Reconfiguration. *Proceedings of the IEEE* **2018**, *106*, 113–128. doi:10.1109/JPROC.2017.2725482.
6. Cheminod, M.; Durante, L.; Valenzano, A. Review of Security Issues in Industrial Networks. *IEEE Transactions on Industrial Informatics* **2013**, *9*, 277–293. doi:10.1109/TII.2012.2198666.
7. George, G.; Thampi, S.M. A Graph-Based Security Framework for Securing Industrial IoT Networks From Vulnerability Exploitations. *IEEE Access* **2018**, *6*, 43586–43601. doi:10.1109/ACCESS.2018.2863244.
8. Fan, X.; Fan, K.; Wang, Y.; Zhou, R. Overview of cyber-security of industrial control system. 2015 International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC), 2015, pp. 1–7. doi:10.1109/SSIC.2015.7245324.
9. Meza, G.; d. Carpio, C.; Vinces, N.; Klusmann, M. Control of a three-axis CNC machine using PLC S7 1200 with the Mach3 software adapted to a Modbus TCP/IP network. 2018 IEEE XXV International Conference on Electronics, Electrical Engineering and Computing (INTERCON), 2018, pp. 1–4. doi:10.1109/INTERCON.2018.8526429.
10. Hittanagi, K.N.; Ramesh, M.; Kumar, K.N.R.; Mahadeva, S.K. PLC based DC drive control using Modbus RTU communication for selected applications of sugar mill. 2017 International Conference on Circuits, Controls, and Communications (CCUBE), 2017, pp. 80–85. doi:10.1109/CCUBE.2017.8394156.
11. Dias, A.L.; Sestito, G.S.; Turcato, A.C.; Brandão, D. Panorama, challenges and opportunities in PROFINET protocol research. 2018 13th IEEE International Conference on Industry Applications (INDUSCON), 2018, pp. 186–193. doi:10.1109/INDUSCON.2018.8627173.
12. Davies, S. Industrial ethernet - The fundamentals of ethernet/IP - EtherNet/IP has reached the million-node landmark, but what is making this protocol so attractive to industrial control engineers? *Computing Control Engineering Journal* **2007**, *18*, 42–45. doi:10.1049/cce:20070110.
13. Denis, M.; Zena, C.; Hayajneh, T. Penetration testing: Concepts, attack methods, and defense strategies. 2016 IEEE Long Island Systems, Applications and Technology Conference (LISAT), 2016, pp. 1–6. doi:10.1109/LISAT.2016.7494156.
14. Shebli, H.M.Z.A.; Beheshti, B.D. A study on penetration testing process and tools. 2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT), 2018, pp. 1–7. doi:10.1109/LISAT.2018.8378035.
15. Ponomarev, S.; Atkison, T. Industrial Control System Network Intrusion Detection by Telemetry Analysis. *IEEE Transactions on Dependable and Secure Computing* **2016**, *13*, 252–260. doi:10.1109/TDSC.2015.2443793.
16. Wan, M.; Shang, W.; Zeng, P. Double Behavior Characteristics for One-Class Classification Anomaly Detection in Networked Control Systems. *IEEE Transactions on Information Forensics and Security* **2017**, *12*, 3011–3023. doi:10.1109/TIFS.2017.2730581.
17. Eigner Oliver, Kreimel Philipp, T.P. Attacks on Industrial Control Systems – Modeling and Anomaly Detection. 2018 - 4th International Conference on Information Systems Security and Privacy, 2018.
18. Keliris, A.; Salehghaffari, H.; Cairl, B.; Krishnamurthy, P.; Maniatakos, M.; Khorrami, F. Machine learning-based defense against process-aware attacks on Industrial Control Systems. 2016 IEEE International Test Conference (ITC), 2016, pp. 1–10. doi:10.1109/TEST.2016.7805855.
19. Mantere, M.; Sailio, M.; Noponen, S. Network Traffic Features for Anomaly Detection in Specific Industrial Control System Network. *Future Internet* **2013**, *5*, 460–473.
20. Jiexin Zhang.; Shaoduo Gan.; Liu, X.; Zhu, P. Intrusion detection in SCADA systems by traffic periodicity and telemetry analysis. 2016 IEEE Symposium on Computers and Communication (ISCC), 2016, pp. 318–325. doi:10.1109/ISCC.2016.7543760.
21. Haripriya, L.; Jabbar, M.A. Role of Machine Learning in Intrusion Detection System: Review. 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2018, pp. 925–929. doi:10.1109/ICECA.2018.8474576.
22. Wagh, S.K.; Kolhe, S.R. Effective intrusion detection system using semi-supervised learning. 2014 International Conference on Data Mining and Intelligent Computing (ICDMIC), 2014, pp. 1–5. doi:10.1109/ICDMIC.2014.6954236.

23. Chaabouni, N.; Mosbah, M.; Zemmari, A.; Sauvignac, C.; Faruki, P. Network Intrusion Detection for IoT Security Based on Learning Techniques. *IEEE Communications Surveys Tutorials* **2019**, *21*, 2671–2701. doi:10.1109/COMST.2019.2896380.
24. Mathur, A.P.; Tippenhauer, N.O. SWaT: a water treatment testbed for research and training on ICS security. 2016 International Workshop on Cyber-physical Systems for Smart Water Networks (CySWater), 2016, pp. 31–36. doi:10.1109/CySWater.2016.7469060.
25. Shah, M.; Soni, V.; Shah, H.; Desai, M. TCP/IP network protocols — Security threats, flaws and defense methods. 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), 2016, pp. 2693–2699.
26. Bobade, S.; Goudar, R. Secure Data Communication Using Protocol Steganography in IPv6. 2015 International Conference on Computing Communication Control and Automation, 2015, pp. 275–279. doi:10.1109/ICCUBEA.2015.59.
27. Ponmaniraj, S.; Rashmi, R.; Anand, M.V. IDS Based Network Security Architecture with TCP/IP Parameters using Machine Learning. 2018 International Conference on Computing, Power and Communication Technologies (GUCON), 2018, pp. 111–114. doi:10.1109/GUCON.2018.8674974.
28. Wei, L.; Gao, D.; Luo, C. False Data Injection Attacks Detection with Deep Belief Networks in Smart Grid. 2018 Chinese Automation Congress (CAC), 2018, pp. 2621–2625. doi:10.1109/CAC.2018.8623514.
29. Wei, M.; Kim, K. Intrusion detection scheme using traffic prediction for wireless industrial networks. *Journal of Communications and Networks* **2012**, *14*, 310–318. doi:10.1109/JCN.2012.6253092.
30. Xiao, Y.; Wang, H.; Xu, W. Parameter Selection of Gaussian Kernel for One-Class SVM. *IEEE Transactions on Cybernetics* **2015**, *45*, 941–953. doi:10.1109/TCYB.2014.2340433.
31. Maglaras, L.A.; Jiang, J.; Cruz, T. Integrated OCSVM mechanism for intrusion detection in SCADA systems. *Electronics Letters* **2014**, *50*, 1935–1936. doi:10.1049/el.2014.2897.
32. Li, Y.; Zhang, T.; Ma, Y.Y.; Zhou, C. Anomaly Detection of User Behavior for Database Security Audit Based on OCSVM. 2016 3rd International Conference on Information Science and Control Engineering (ICISCE), 2016, pp. 214–219. doi:10.1109/ICISCE.2016.55.
33. Keerthi, S.S.; Shevade, S.K.; Bhattacharyya, C.; Murthy, K.R.K. Improvements to Platt's SMO Algorithm for SVM Classifier Design. *Neural Computation* **2001**, *13*, 637–649. doi:10.1162/089976601300014493.
34. Toyoda, K.; Okamoto, T.; Koakutsu, S. An optimal routing search method on the network routing problem using the sequential minimal optimization. 2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), 2017, pp. 805–810. doi:10.23919/SICE.2017.8105653.
35. Sheenu.; Joshi, G.; Vig, R. A multi-class hand gesture recognition in complex background using Sequential minimal Optimization. 2015 International Conference on Signal Processing, Computing and Control (ISPCC), 2015, pp. 92–96. doi:10.1109/ISPCC.2015.7375004.
36. Khraisat, A.; Gondal, I.; Vamplew, P.; Kamruzzaman, J. Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity* **2019**, *2*, 1–22. doi:10.1186/s42400-019-0038-7.
37. Hindy, H.; Brosset, D.; Bayne, E.; Seeam, A.; Tachtatzis, C.; Atkinson, R.C.; Bellekens, X.J.A. A Taxonomy and Survey of Intrusion Detection System Design Techniques, Network Threats and Datasets. *CoRR* **2018**, *abs/1806.03517*, [1806.03517].