Feasibility of Using Seq-GAN Model in Vulnerability Detection of Industrial Control Protocols

Jiafa Zhang*, Hong Zou, Zifeng Zeng, Weijie Xu and Jiawei Jiang

China Southern Power Grid Digital Grid Group Information and Communication Technology Co., Ltd, Guangzhou City, Guangdong Province, China 510663
E-mail: zengzf@csg.cn
**Corresponding Author*

Received 08 October 2023; Accepted 11 January 2024;
Publication 09 April 2024

Abstract

Continuous improvement of internet technology has driven the continuous progress and improvement of industrial control systems, and provided more support for security vulnerability detection in this field. Combining the GAN model, the detection model based on Seq-GAN in industrial control protocol vulnerabilities constructed in this article provides more options for further improving the security of industrial control systems, and can detect and analyse security vulnerabilities in industrial control protocols more efficiently and accurately. By comparing the performance of different models for security vulnerability detection, the Seq-GAN model has smaller prediction errors, can also obtain higher G-mean and F1-score values, and has sufficient reliability. At the same time, it can also improve the efficiency of vulnerability

Journal of Cyber Security and Mobility, Vol. 13_3, 393–416.
doi: 10.13052/jcsm2245-1439.1333

detection in industrial control systems, and can achieve better comprehensive detection performance. Therefore, the application of the Seq-GAN model in industrial control protocol vulnerability detection can provide more support for improving security detection in this field.

Keywords: Seq-GAN, industrial safety, control protocol, vulnerability detection, feasibility.

1 Introduction

In the field of manufacturing, industrial software is the cornerstone of industrial production, providing support for the rapid development of industrial production [1, 2]. Industrial software, as an important component of industrial survival, not only helps industrial production become more convenient and efficient, but also improves the efficiency of industrial production. As a fundamental component of building the industrial internet, the security of industrial software is directly related to the safe, stable, and reliable operation of industrial generation, so its security issues are very important.

In the process of continuous improvement and updating of industrial software, the development of Industrial Control System (ICS) is also gradually improving. The industrial control system connects industrial control software and hardware with the network, mainly responsible for real-time data collection, data system monitoring, and automatic control and management during industrial production processes. The industrial control system consists of two parts, and the Operational Technology (OT) network is used for monitoring and managing industrial equipment [3, 4]. Information technology (IT) includes databases, workstations, and traditional machines that process information. The communication protocol of industrial control systems plays an important role in circuit communication and control in industrial control systems and industrial software. Since industrial control protocol information can directly or indirectly transmit the operation status or operation of industrial software, there are potential safety risks. The development of informatization has led to the interconnection of IT/OT networks, simplifying the digital operation process, and inevitably creating vulnerabilities that are vulnerable to attacks, posing more threats to the system [5]. Attacks against industrial control protocols and traditional industrial software security vulnerabilities have made industrial production and manufacturing the target of numerous attackers.

The security of industrial control systems is manifested at different levels, such as ensuring the reliability of embedded programs through formal verification; Through penetration testing to identify system vulnerabilities, etc. Different research fields and related achievements have improved the safety of industrial control systems to varying degrees [6, 7]. However, compared to traditional manufacturing, intelligent manufacturing has a significant new feature – increasing interconnectivity between industrial control systems, and more complex and diverse internal control systems. With the rapid development of industrial informatization, the interaction between different subsystems in industrial control protocols has become increasingly frequent. Industrial control protocols serve as communication bridges between various parts of industrial control systems. Industrial control protocols always have some inherent vulnerabilities or issues. If such internal security vulnerabilities cannot be detected and fixed in a timely manner, they will become important targets for attacks against industrial control protocols [8].

In the face of vulnerabilities in industrial control protocols and the detection and analysis of vulnerabilities, generative adversarial networks (GANs) provide a more suitable detection method with better performance. The generative adversarial network is a groundbreaking framework that guides the training of generative models in the direction of unsupervised training. As in game theory, the two networks of generators and decision makers manipulate each other, deceiving each other while making them stronger, ultimately achieving a certain balance [9, 10]. In GAN, a discriminator network D (Discriminator) learns to distinguish between the real situation of a given data instance: whether the given judgment example comes from a real example in the database or a pseudo example generated by the generator; The generative network G (Generator) confuses the interference discriminator's decision ability by learning to generate false but highly reliable data. In many generation tasks that replicate rich content from the real world, generative adversarial networks exhibit significant advantages. In industrial control protocol vulnerability detection, this feature can be utilized to generate false but pseudo real sequence information. Meanwhile, compared to other deep learning models, GAN based training models have certain instability, and the main reason for the difficulty in training is that the architecture not only involves simultaneously training generators and decision models in zero sum games, but also involves the requirement to ensure that the learning abilities of the two neural networks can be synchronized and balanced [11–13].

Based on the above analysis, this article combines generative adversarial networks in deep learning and reinforcement learning ideas to design and implement a more efficient detection model for vulnerability detection in industrial control protocols. This model addresses the problem of difficult generation of discrete data in traditional generative adversarial networks. By utilizing traditional GAN networks and incorporating reinforcement learning ideas, a Sequential Generative Adversarial Network (Seq-GAN) model is constructed. Through the application of this model in industrial control protocol vulnerability detection, the diversity of vulnerability detection and acceptance rate of test cases in industrial control protocols are further improved, this lays the foundation for the feasibility of applying the Seq-GAN model in vulnerability detection of industrial control protocols.

2 Sequence Generation Theory Based on GAN Networks

2.1 Generative Adversarial Neural Network

In generative adversarial networks (GAN), the main function of generators is to generate simulation data that is similar to the original real data. The GAN model is mainly composed of two neural networks, the generator model and the discriminator model, which are combined to learn through game theory.

The generator model obtains a random vector as the input value for relevant calculations and analysis, and after a series of data calculations and transformations, provides new transformed data. The main function of the discriminator model is to determine and distinguish the authenticity of the data transmitted by the generator. The above two models are continuously optimized and learned through this adversarial approach. In the conventional adversarial learning process, the samples generated by the generator model are input into the discriminator for analysis and judgment. The discriminator model outputs a probability value based on the received data after calculation and analysis, which represents the likelihood that the sample data is real data. In the relevant process, the main goal of the generator model is to generate a series of data information that can deceive the discriminator, so that the generated data can interfere with the discriminator. Through training on this interference, the discriminator model cannot distinguish whether the data is real or generated. Meanwhile, the discriminator model in the GAN model is based on the received data for analysis and judgment, accurately determining the authenticity of the data input by the generator. During the training process, the two models play games with each other and continuously improve their

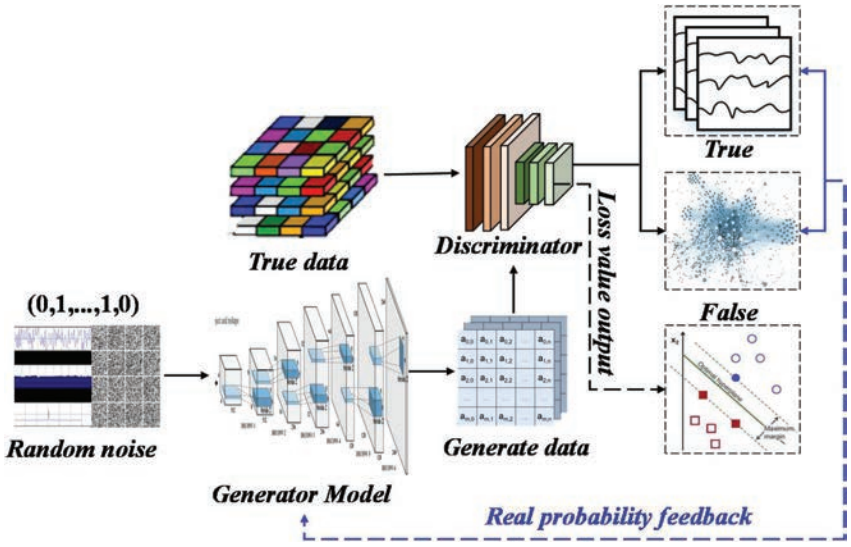


Figure 1 Structure diagram of generative adversarial neural network model.

abilities until a certain training effect is achieved [14–16]. As shown in Figure 1, a structural diagram for generating an adversarial network model is provided.

Through the continuous process of adversarial learning mentioned above, the generator model continuously optimizes the generated data, making the generated data more similar to the real data, and the interference to the accurate judgment of the discriminator is becoming stronger and stronger. For the discriminator model, it is also continuously improving its ability to determine the authenticity of the data generated by the generator. The final implementation result is that the generator model can generate new data similar to the original data, while the discriminator model can accurately determine the authenticity of the data [17]. The two constantly play games with each other, achieving continuous optimization and improvement of the generator model and discriminator model, ultimately achieving equilibrium. The specific model optimization process is shown in formula (1):

$$\underset{G}{min}\ \underset{D}{max}\ V(D,G) = E_{X\sim Pdata(x)}[\log D(x)] + E_{Z\sim PZ(z)}\{\log(1-D[G(x)])\} \tag{1}$$

In the equation, H is the generator model, Z is a random variable, $G(Z)$ is the data generated by the generator after inputting the random

variable. $X \sim Pdata\ (x)$ is the distribution of real data. $Z \tilde{} PZ(z)$ is the Gaussian distribution of random data.

2.2 Sequence Generation Based on GAN Network

In response to the limitations faced by the GAN algorithm, a sequence generation adversarial network (Seq-GAN) model was constructed by introducing a sequence generation algorithm. This model is based on a combination of generative adversarial network (GAN) and reinforcement learning (RL) for sequence generation, which can be used to generate discrete data sequences. Solved the problem of traditional GAN networks being unable to transfer discrete data from discriminators to generators during training [18, 19]. To address the issue of the discriminator model not being able to evaluate complete sequences, Seq-GAN used Monte Carlo search in policy gradients to further analyse and evaluate sequence generation. In the Seq-GAN model, the generator model acts as an agent and generates an element at each step, forming a sequence that is considered a state in reinforcement learning. The next element to be generated is considered an action in reinforcement learning. The generator model generates the next state and action based on the current state and action, until the end of the sequence. The discriminator model serves as the environment, receiving the sequence generated by the generator and discriminating against it, thereby generating a reward signal, and transmitting it to the generator model. The generator model updates its own parameters based on reward signals to better generate the next sequence [20]. The basic idea of Seq-GAN originates from GAN networks, which generate high-quality discrete data such as speech sequences, text sequences, and time series by incorporating reinforcement learning ideas. As shown in Figure 2, a schematic diagram of the network structure of the Seq-GAN network is provided.

The Seq-GAN network model also includes generators and discriminators. Due to the discrete nature of the sequence, it is usually not possible to directly pass the gradient update generator parameters back from the discriminator. Therefore, the Seq-GAN network incorporates reinforcement learning ideas, treating the Policy network in reinforcement learning as generator G, treating existing dots (a dot representing a word or a word) as the current state, and the next dot to be generated as an action. At this point, the sequence is generated word by word, and incomplete sequences cannot be input into the discrimination for recognition and judgment [21, 22]. Therefore, when determining the action at the next moment, it is necessary to complete the

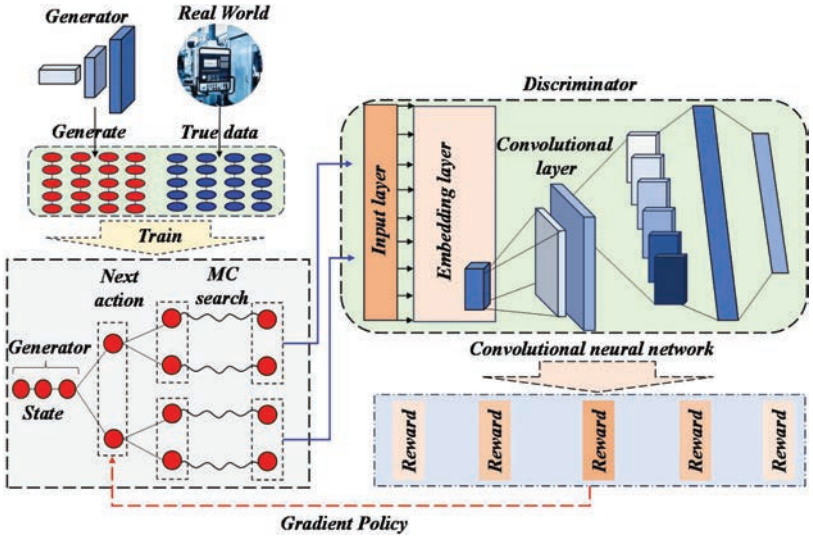


Figure 2 Schematic diagram of Seq-GAN network structure.

incomplete sequence and then input it into the discriminator. The specific approach is to use the Monte Carlo tree search algorithm to generate all possible complete sequences for each action. Discriminator D generates a Reward by identifying these complete sequences and transmitting it back to generator G, thereby completing the generator's update at the current time. The next time, the state will be updated until a complete sequence is generated.

3 Construction of Industrial Control Protocol Vulnerability Detection Based on Seq-GAN Model

3.1 Theory of Seq-GAN Model

Generative adversarial networks can generate data that is equally distributed as real data, and this feature can be used to generate desired content. However, generative adversarial networks have its limitations, as they can only generate the distribution of continuous data. Once the data is discrete, generative adversarial networks cannot be processed, thus making it difficult for the discriminator to make better judgments. So there needs to be a method that can both reflect changes and be applicable to discrete data. Seq-GAN cleverly utilizes the method of strategy gradient to treat the process of sequence

generation as a series of decision-making processes [23]. This allows us to "infer" what the next generated node should be based on the already generated parts, thereby generating the entire sequence. Then, the entire sequence is tested by the discriminant network and the decision result of the sequence is also obtained, which is very similar to GAN.

The training process of Seq-GAN usually consists of two parts: first, pre-training, which mainly includes using the maximum likelihood method to train the generated network; Then there is adversarial-training, which uses the data generated by the generated network as negative samples to train the discriminative network. The purpose of this step is to reduce the error range of the network to a range that can effectively converge before subsequent adversarial training, in order to prevent direct use of adversarial training from causing the direction of network convergence to be uncontrollable, resulting in completely unsatisfactory results. The training of discriminative networks in adversarial training is similar to the training of discriminative networks in pre training, with the entire sequence as the discriminative object, making the discrimination more accurate. The generative network requires the construction of a parameter shared roll out extension network, which is responsible for starting from a certain position in the sequence generated by the generative network and succeeding the generative network by generating [24]. Because in adversarial training, the training of the generative network is a Monte Carlo tree search, which takes the generated sequence as the state, takes any position from the beginning as the current state, and selects the next point as the behaviour. As shown in Figure 3, a sequence generation adversarial network training method is provided.

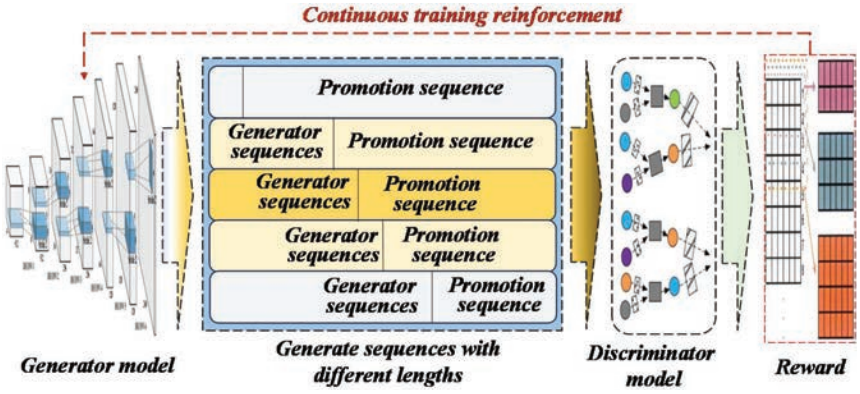


Figure 3 Seq-GAN training process.

3.2 Construction of a Vulnerability Detection Model Based on Seq-GAN

3.2.1 Construction of vulnerability detection model for industrial control protocol

Seq-GAN mainly draws on reinforcement learning methods. The generator's task $G(\theta)$ is to generate a sequence $Y_{1:T} = \{y_1, y^2, \ldots, y_{t-1}\}$ under the neural network parameter 0. At time *t*, the state *s* is the generated sequence $\{y_1, y_2, \ldots, y_{t-1}\}$, and the action a is the next data y_t to be generated. Currently, the generator model is $G_\theta(y_t|Y_{1:T-1})$. Formula (2) provides a description of the optimal objective function:

$$J(\theta) = E(R_T|s_0, \theta) = \sum_{y_1 \in Y} G_\theta(y_1|S_0) Q_{D_\theta}^{G_\theta}(s_0, y_1) \tag{2}$$

The formula can be explained as the expectation of generating a complete sequence under the conditions of S_0 and θ [25]. $Q_{D_\theta}^{G_\theta}(s_0, y_1)$ is the action value function obtained from the discriminator model, and its calculation process is shown in formula (3):

$$Q_{D_\theta}^{G_\theta}(s = Y_{1:t-1}, a = y_T) = D_\theta(Y_{1:T}) \tag{3}$$

The Monte Carlo algorithm can be used to evaluate the rewards passed by the discriminator to the generator. After *N* rounds of Monte Carlo algorithm, the sequence can be obtained as shown in formula (4):

$$[Y_{1:T}^1, \ldots, Y_{1:T}^N] = MC^{G_\beta}(Y_{1:t}^1; N) \tag{4}$$

At this point, the G_β and generator settings are the same. The generator model G_β and Monte Carlo algorithm generate *N* candidate options and generate new states based on the output of the generated model. For each candidate option, the discriminant function $D_\theta(Y_{1:T})$ scores them and evaluates them using a value network. The total score of the candidate options is calculated, which is the average value of the value network evaluation and the discriminant function score [26, 27]. The calculation formulas (5) and (6) are shown. When the last time step of the action is $t = T$, the discriminant function $D_\theta(Y_{1:T})$ is directly used to calculate the final estimate as the score of the state.

$$Q_{D_\theta}^{G_\theta}(s = Y_{1:t-1}, a = y_T) = \frac{1}{N} \sum_{n=1}^{N} D_\theta(Y_{1:T}^n) \quad (t < T) \tag{5}$$

$$Q_{D_\theta}^{G_\theta}(s = Y_{1:t-1}, a = y_T) = D_\theta(Y_{1:T} \quad (t = T) \tag{6}$$

At this point, use the generated data to retrain *D*, as shown in formula (7):

$$\min_{\theta} -E_{\gamma-Pdata}[\log D_\theta(Y)] - E_{\gamma-G_\theta}[\log(1 - D_\theta(Y))] \tag{7}$$

Minimize the inverse number of *D* determining true data and *D* generating false data. After training one or more rounds of *D*, use Policy gradient to update *G*, as shown in formula (8):

$$\nabla J(\theta) = \sum_{t=1}^{T} E_{Y_{1:T-1}\sim G_\theta} \left[\sum_{y_t \in V} \nabla_\theta G_\theta(y_t|Y_{1:t-1}) \times D_{D_\theta}^{G_\theta}(Y_{1:t-1}, y_t) \right] \tag{8}$$

At this point, the parameters of the generator need to be modified according to the gradient, as shown in formula (9):

$$\theta = \theta + \alpha_h \nabla_\theta J(\theta) \tag{9}$$

Recurrent neural networks can fully learn time and input sequence information by analyzing input information. The LSTM used in this article as a generator model has more obvious advantages, mainly manifested in its composition of embedding layer, LSTM layer, perception layer, and *softmax*. The embedding layer transforms the input sequences $\{y_1, y_2, \ldots, y_t\}$, encodes each character into a vector, and uses the sequence data at the LSTM layer to "Remember" the relationships between the sequences. The update function g is applied to convert the sequence into hidden vectors $\{h_1, h_2, \ldots, h_t\}$, formula (10) is ht. Specific update formula:

$$h_t = g(h_{t-1}, x_t) \tag{10}$$

In addition, the *softmax* output layer z obtains the output vector by weighting and normalizing the hidden states, as shown in formula (11):

$$p(y_t|x_1, \ldots, x_t) = z(h_t) \tag{11}$$

For $z(h_t)$, formula (12) can be used to describe:

$$z(h_t) = softmax(c + V \times h_t) \tag{12}$$

In the equation, c is the bias of the *softmax* function, and *V* is the weight matrix. In order to address the problem of gradient vanishing and gradient explosion, the LSTM update function g is used to update, adding a fixed h_0 to each loop unit and updating the constant.

3.2.2 Industrial control protocol vulnerability detection method and process

The industrial control protocol vulnerability detection model based on the Seq-GAN model constructed in this article mainly uses the generation of adversarial networks as the basic theory to generate traffic data in industrial control network systems [28]. This method can intelligently learn the format of native data frames and generate similar format data frames for attacking the target system. By mining many communication data patterns, a specific generation model can be obtained to generate similar industrial control system communication data. Then, the generated data is sent to the system to be tested and abnormal behaviour generated by the system is recorded to discover system vulnerabilities. Because the traffic data in industrial control systems exist as sequences and have many similarities, this method can theoretically be used in most industrial control systems to improve the challenges faced by vulnerability detection in industrial control protocols [29]. As shown in Figure 4, the basic process of this method detection is presented.

By analysing the Seq-GAN based industrial control protocol vulnerability detection model, it can be found that the process involved in this model is composed of three main processes. Firstly, the construction of the

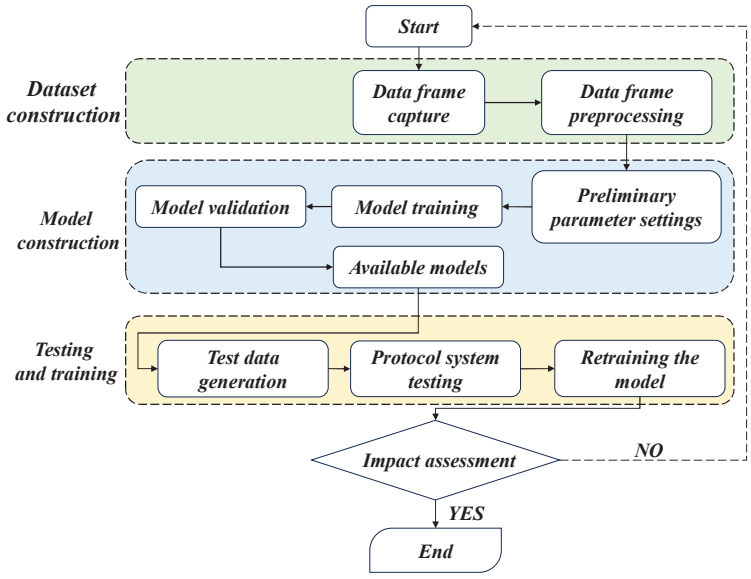


Figure 4 Process of industrial control protocol vulnerability detection model based on Seq-GAN.

dataset: The training and formation of adversarial network models require a large amount of data, and the construction of the dataset needs to be completed before the start of model training to ensure its rationality. In actual industrial control systems, a large amount of communication data can be captured. Finally, the processed data will be composed into a dataset for training the model. Secondly, the construction of the model: the neural network model can output corresponding outputs by giving a certain input, so it can be regarded as a function. On the premise of obtaining effective fuzzy test data, considering other factors, the model can consume less computation, reduce time, and improve efficiency. After the model is trained, it indicates that the model has achieved effectiveness in the training data, but the goal of the model is to use new data for application. Finally, attack testing and model retraining: using the constructed model to generate fuzzy test data for attack testing on the target system, and attacking the target system to discover vulnerabilities [30, 31]. In order to improve the method capability and enhance its usability in practice, a retraining model was added after fuzzy testing of the target system. The data frames that caused anomalies in the target system were collected and mutated to retrain the model, further improving its performance in industrial control protocol vulnerability detection.

4 Model Experiment and Result Analysis

4.1 Model Evaluation Indicators

There is no comprehensive and reasonable evaluation index in the field of application performance testing in industrial control protocol vulnerability detection based on the Seq-GAN model. In order to compare and analyse the performance of the model, this article compares the performance of the model used for industrial control protocol test case generation from the dimensions of accuracy, recall, accuracy, G-mean, and F1-score. The calculation process of Precision is shown in formula (13):

$$P = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N}(TP_i + FP)} \tag{13}$$

Recall rate represents the proportion of correctly detected negative samples to all negative samples, and the calculation process is shown in formula (14):

$$R = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N}(TP_i + FN_i)} \tag{14}$$

Accuracy represents the proportion of correctly detected negative and positive samples to the total number, and the calculation process is shown in formula (15):

$$A = \frac{\sum_{i=1}^{N}(TP_i + TN_i)}{\sum_{i=1}^{N}(TP_i + FN_i + TN_i + FP_i)} \tag{15}$$

For the G-mean indicator, it represents the geometric mean of specificity and recall, as shown in formula (16):

$$G\text{-}mean = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}} \tag{16}$$

In addition, F1-score is the harmonic mean of accuracy and recall, commonly used to comprehensively evaluate the accuracy of classification models. The specific calculation process is given by formula (17):

$$F1\text{-}score = \frac{2P \times R}{P + R} \tag{17}$$

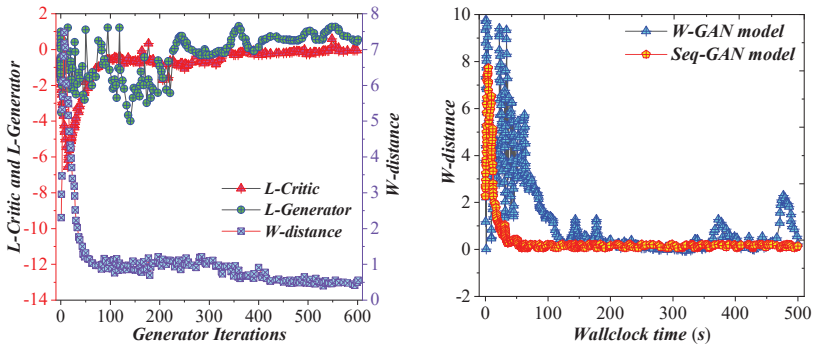
In addition, combined with the application of the Seq-GAN model in industrial control protocol vulnerability detection, this article further analyses indicators such as Test Input Acceptance Rate (TIAR) and Ability of Vulnerability Detection (AVD) of the model. TIAR reflects the proportion of all data received by the test target, and the more data is accepted, the greater the likelihood of vulnerabilities occurring in the test target. AVD is ability to trigger system anomalies and is a direct indicator of the effectiveness of evaluation methods, as our goal is to trigger as many system anomalies as possible and discover vulnerabilities. The calculation of test case acceptance rate and vulnerability detection ability is given by formulas (18) and (19):

$$TIAR = \frac{n_{Accept}}{n_{Cent}} \times 100\% \tag{18}$$

$$AVD = \frac{n_{Bugs}}{n_{Cases}} \times 100\% \tag{19}$$

4.2 Model Test Results

After collecting the dataset required for model training, it is necessary to preprocess the dataset; Next, training the model designed in the previous section by using the data from the training set. The advantage of the Seq-GAN model is that it largely overcomes the instability problem of GAN training.



a) Comparison of Wasserstein Distances under Different Epochs

b) Comparison of Wall Clock Time under Different Epochs

Figure 5 Comparative analysis of Wasserstein distance and wall clock time under different epochs.

This model is based on GAN architecture constraints to design and improve its own model architecture, and utilizes distance and gradient penalties to improve training speed and sample quality. To further analyse its performance changes in industrial control protocol vulnerability detection, the Seq-GAN model was trained on a dataset with 500000 test cases in the experiment. As shown in Figure 5, the Seq-GAN distance and the loss function values of the generator and evaluator are given. For comparative analysis, this article trained another W-GAN based model using the same hyperparameter setting. The comparison results show that the industrial control protocol vulnerability detection method based on the Seq-GAN model not only has faster convergence speed, but also a more stable training process.

After the model training is completed, use the generator of the trained model to generate pseudo test cases, input them into the tested industrial control system, and implement fuzzy testing for industrial control protocols according to the standard fuzzy testing process. During the process of inputting test cases into the industrial control system under test, it is necessary to monitor the system in real time, record output faults or interface crashes, and restart the testing process at the current point. At the end of the entire test, by analysing the causes of previous anomalies and the logs of the communication server system, the operational status of potential vulnerabilities can be identified.

During the testing phase, more attention is paid to the recall rate, while also considering the G-mean index and F1-score. Therefore, this experiment

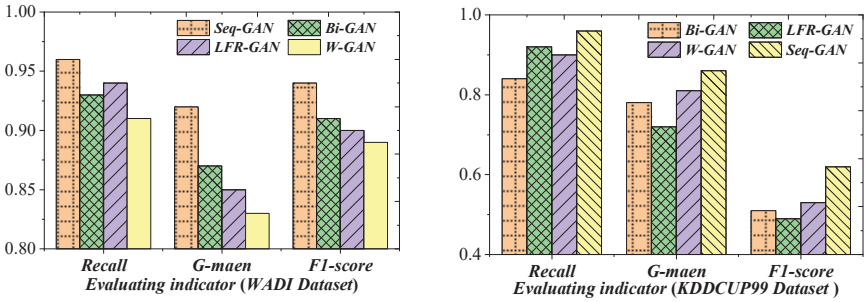


Figure 6 Comparison of detection results of different models under different datasets.

uses recall, G-mean, and F1-score to reflect the detection performance of the model. F1-score is the harmonic average of recall and accuracy, and G-mean is the geometric average of specificity and recall, both of which can reflect the comprehensive performance of the model. Based on the above main performance indicators, a comparative analysis was conducted between the Seq-GAN based industrial control protocol vulnerability detection model and the changes in indicators of other models. The detection results are shown in Figure 6.

As shown in the figure, the Seq-GAN model has a recall rate of 0.96 on the WADI dataset and 0.98 on the KDDCUP99 dataset. Compared to other models, the recall rate of the Seq-GAN model can be improved by approximately 2% to 7%. The increase in recall rate represents a decrease in the false alarm rate generated by the model for abnormal data. The Seq-GAN model can not only more accurately identify abnormal samples, improve system security, but also consider the real-time performance of the system, avoiding situations where the system response is invalid due to high false alarm rates. In addition, in the detection of the WADI dataset, the G-mean and F1-score indicators of the Seq-GAN based industrial control protocol vulnerability detection model improved by 5% to 9% and 4% to 8% compared to other comparative models, respectively. Through the above experimental results, it can be found that compared to other models, the Seq-GAN based industrial control protocol vulnerability detection model constructed in this paper has better detection performance and can achieve more accurate detection results.

There are certain differences in the impact of different sampling models on the classification confidence of classifiers. In order to quantitatively analyse the impact of different models on classifiers, this article combines the characteristics of different models and selects reliability graphs to discuss the rationality of the model's class prediction probability. In the reliability

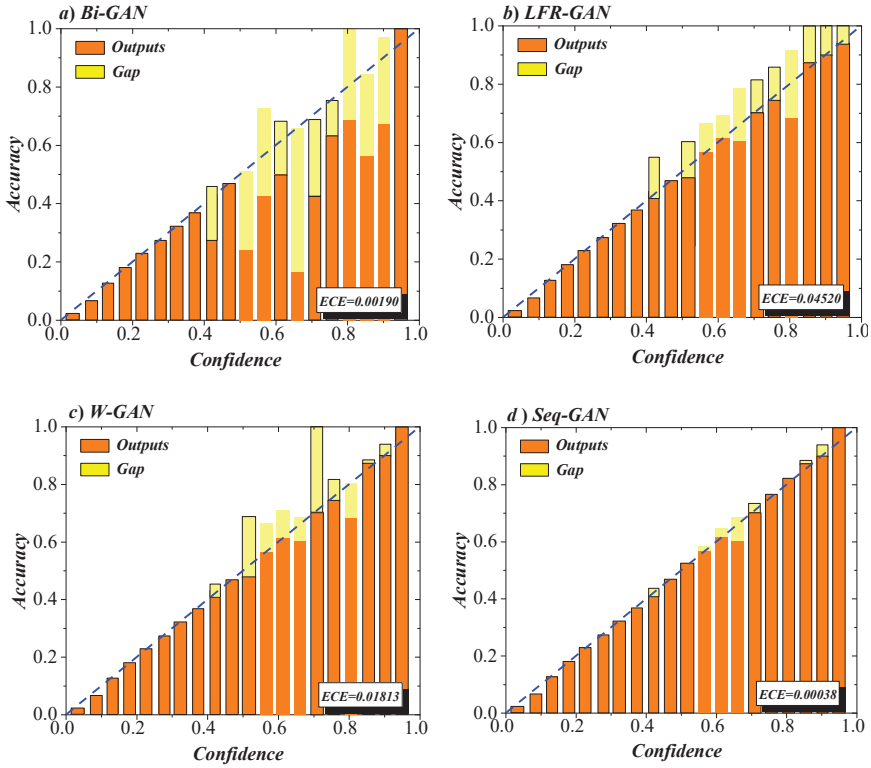


Figure 7 Reliability analysis of various sampling methods on KDDCUP99 dataset.

analysis chart, the blue dashed line is the optimal calibration. In addition, ECE was selected as the measure of model confidence when analysing relevant indicators in this experiment. The closer the accuracy and confidence, the smaller the obtained ECE value, indicating a higher confidence level of the model. At this point, the dataset used in the experiment was KDDCUP99. As shown in Figure 7, experimental results of reliability analysis for different models are presented. It can be seen from the figure that the industrial control protocol vulnerability detection model constructed in this paper has higher reliability.

After the model training is completed, to ensure the feasibility of the new data, further validation is mainly carried out on the Seq-GAN model. Through relevant analysis, it can be found that there are significant differences in the performance of different models on TIAR indicators when the learning rates are 0.02 and 0.001. When the learning rate is set to 0.001, TIAR can

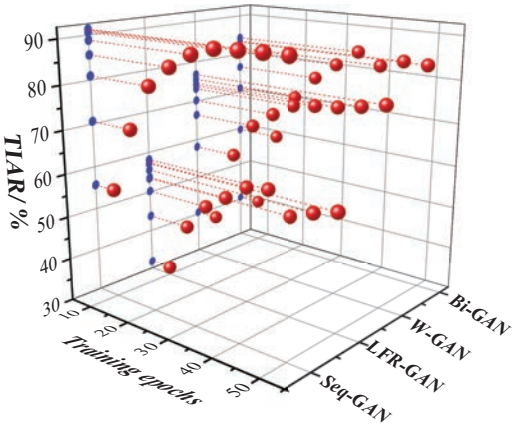


Figure 8 Comparison of test input accept rate for different models.

steadily improve without fluctuations, so using a learning rate of 0.001 in the experiment is appropriate. As shown in Figure 8, the comparison of TIAR indicators when using Modbus slave as the attack target is presented. As shown in the figure, as the training cycle increases, the TIAR value gradually increases, indicating that more and more generated data meets the acceptance requirements of the testing objectives. If the test data is accepted, it indicates that the accuracy of the generated data format is increasing. Compared with other models, the Seq-GAN model can achieve higher TIAR. The highest point of the Seq-GAN model reached 91%, indicating that a small portion of the data was not formatted correctly. However, in terms of data acceptance rate comparison, the Seq-GAN model has more advantages in industrial control protocol vulnerability detection.

In addition, the AVD (Ability of Vulnerability Detection) indicators that evaluate the vulnerability detection capabilities of different models can more intuitively reflect the performance of the model in vulnerability detection in industrial control protocols. As shown in Figure 9, in the experiment, Modbus slave was used as the test target, and the changes in AVD indicators between different models and Seq-GAN models were presented. As shown in the figure, as the training time of AVD gradually increases, it indicates that the number of errors caused by generated data is increasing, and the AVD curve ultimately reaches a flat stage and reaches its highest point. The height that can be achieved in practice is not only related to the method itself, but also to the testing objectives. If the tested target contains many vulnerabilities and defects, the final peak will also be larger. The Seq-GAN model

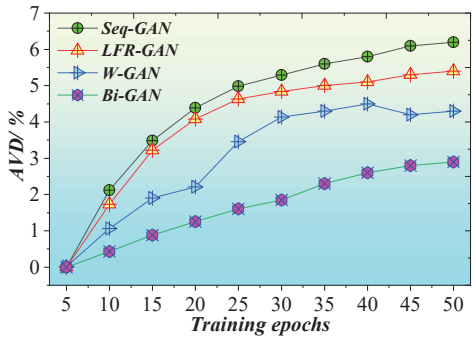


Figure 9 Ability of vulnerability detection efficiency of different models.

has greater advantages compared to other models in terms of vulnerability detection performance in industrial control protocols.

5 Conclusions

Internet technology has promoted the continuous improvement of security vulnerability detection technology in the field of industrial control, providing more convenient means for the detection and analysis of vulnerabilities in industrial control protocols. Based on the application of GAN model in the field of security vulnerability detection, this article constructs a Seq-GAN based industrial control protocol vulnerability detection model, and compares and analyses the relevant performance indicators with experimental data, summarizing the application advantages of this model in the field of industrial control protocol vulnerability detection. The main conclusions drawn are as follows:

(1) Combining the GAN model, the Seq-GAN based industrial control protocol vulnerability detection model constructed can achieve better overall performance compared to other models, achieving further improvement in security vulnerability detection performance. Through experimental analysis, it can be concluded that the error of this model in industrial control protocol vulnerability detection is relatively low; The highest recall rate of the model is 96%, which can be improved by 2% to 7%. At the same time, the G-mean and F1-score indicators can also be significantly improved. The reliability analysis results also indicate that the model has better evaluation results.

(2) Through analysis of its TIAR and AVD indicators, when the Training epoch is 50, the test case pass rate of the Seq-GAN model is about 91%, which is significantly better than other models; In terms of its vulnerability detection efficiency, the Seq-GAN model has an improvement range of 0.7% to 3.5%, which is significantly improved compared to other models. It exhibits more advantageous comprehensive detection performance in industrial control protocol vulnerability detection, providing support for security vulnerability detection and analysis in the field of industrial control.

References

[1] Lasi H, Fettke P, Kemper H-G, et al. Industry 4.0[J]. Business & information systems engineering, 2014, 6(4):239–242.

[2] Alzahrani A, Aldhyani H H T. Design of Efficient Based Artificial Intelligence Approaches for Sustainable of Cyber Security in Smart Industrial Control System[J]. Sustainability, 2023, 15(10):162–139.

[3] Lamshöft, K., Hielscher, J., Krätzer, C., and Dittmann, J. The Threat of Covert Channels in Network Time Synchronisation Protocols. Journal of Cyber Security and Mobility, 2022, 11(02), 165–204.

[4] Yaofang Z, Zibo W, Yingzhou W, et al. A risk assessment model for similar attack scenarios in industrial control system[J]. The Journal of Supercomputing, 2023, 79(14):955–979.

[5] Mohammed B, Ahmed A, Yaser D, et al. LPWAN Cyber Security Risk Analysis: Building a Secure IQRF Solution[J]. Sensors, 2023, 23(4): 59–79.

[6] Meng Z, Xie Y, Sun J. Detecting Credit Card Fraud by Generative Adversarial Networks and Multi-head Attention Neural Networks[J]. IAENG International Journal of Computer Science, 2023, 50(2):42–59.

[7] Xue, Y. Machine Learning: Research on Detection of Network Security Vulnerabilities by Extracting and Matching Features[J]. Journal of Cyber Security and Mobility, 2023, 12(5), 697–710.

[8] Rowan T H, Liming Z, Tomasz B. Generative Adversarial Networks–Enabled Human–Artificial Intelligence Collaborative Applications for Creative and Design Industries: A Systematic Review of Current Approaches and Trends[J]. Frontiers in Artificial Intelligence, 2021, 4(6):132–148.

[9] John C. Industrial control system risk assessment standards and leading practices in the chemical industry[J]. Process Safety Progress, 2022, 41(4):665–669.
[10] Li E, Kang C, Huang D, et al. Quantitative Model of Attacks on Distribution Automation Systems Based on CVSS and Attack Trees[J]. Information, 2019, 10(8):215–251.
[11] Salamon J, Bello J P. Deep convolutional neural networks and data augmentation for environmental sound classification[J]. IEEE Signal Processing Letters, 2017, 24(3): 279–283.
[12] Jili T, Yang B, Zhuoyi C, et al. Improved constrained min-max optimization for MPC tracking control in industrial networked process systems[J]. Measurement and Control, 2023, 56(1–2):114–123.
[13] Dari, E. Y., Bendahmane, A., and Essaaidi, M. Optimal Method for Detecting Collusive Saboteur Smart Meters in Smart Grid[J]. Journal of Cyber Security and Mobility, 2020, 9(2), 237–264.
[14] Lv Z. A novel LightGBM-based industrial internet intrusion detection method[J]. International Journal of Computer Applications in Technology, 2023, 71(3):208–216.
[15] Lv W, Xiong J, Shi J, et al. A deep convolution generative adversarial networks based fuzzing framework for industry control protocols[J]. Journal of Intelligent Manufacturing, 2020, 32(2):1–17.
[16] Yun S, Zhaoyu C, Xuejun L, et al. Adaptive Industrial Control System Attack Sample Expansion Algorithm Based on Generative Adversarial Network[J]. Applied Sciences, 2022, 12(17):102–121.
[17] Mingyu K, Ran Z, Duxin C, et al. CM-GAN: A Cross-Modal Generative Adversarial Network for Imputing Completely Missing Data in Digital Industry.[J]. IEEE transactions on neural networks and learning systems, 2023.
[18] Zheng M, Li T, Zhu R, et al. Conditional Wasserstein generative adversarial network-gradient penalty-based approach to alleviating imbalanced data classification[J]. Information Sciences, 2020, 5(12): 1009–1023.
[19] Kim K. GAN based augmentation for improving anomaly detection accuracy in host-based intrusion detection systems[J]. Int. J. Eng. Res. Technol, 2020, 13: 3987.
[20] Wei G, Yijin W, Xin C, et al. Federated transfer learning for auxiliary classifier generative adversarial networks: framework and industrial application.[J]. Journal of intelligent manufacturing, 2023.

[21] Jin R, Niu Q. Research on textile defects detection based on improved generative adversarial network[J]. Journal of Engineered Fibers and Fabrics, 2022, 17.

[22] Mingwei Z, Min Z, Min Y, et al. Generative Adversarial Network of Industrial Positron Images on Memory Module[J]. Entropy, 2022, 24(6):793–805.

[23] Purandhar N, Ayyasamy S, Kumar P. Classification of clustered health care data analysis using generative adversarial networks (GAN)[J]. Soft Computing, 2022, 26(12):511–521.

[24] Vagan T, Svitlana G, Mariia G. Taxonomy of generative adversarial networks for digital immunity of Industry 4.0 systems[J]. Procedia Computer Science, 2021, 180.

[25] Sohn K, Sung E C, Koo G, et al. Artificial intelligence in the fashion industry: consumer responses to generative adversarial network (GAN) technology[J]. International Journal of Retail & Distribution Management, 2020, 49(1):61–80.

[26] He R, Li X, Chen G, et al. Generative adversarial network-based semi-supervised learning for real-time risk warning of process industries[J]. Expert Systems With Applications, 2020, 150.

[27] Szubert B, Cole J E, Monaco C, et al. Structure-preserving visualisation of high dimensional single-cell datasets[J]. Scientific reports, 2019, 9(1):1-10.

[28] Oh E, Lee H. An Imbalanced Data Handling Framework for Industrial Big Data Using a Gaussian Process Regression-Based Generative Adversarial Network[J]. Symmetry, 2020, 12(4):46–67.

[29] Zhang K, Zhang Y, Cheng H. CrackGAN: A Labor-Light Crack Detection Approach Using Industrial Pavement Images Based on Generative Adversarial Learning.[J]. CoRR, 2019, abs/1909.08216.

[30] Liu X, Li T, Zhang R, et al. A GAN and Feature Selection-Based Oversampling Technique for Intrusion Detection[J]. Security and Communication Networks, 2021.

[31] Yue G, Ping G, Lanxin L. An End-to-End model based on CNN-LSTM for Industrial Fault Diagnosis and Prognosis[C] //International Conference on Network Infrastructure and Digital Content (IC-NIDC). 2018: 274–278.

Biographies

Jiafa Zhang (May 1985) male, Han nationality, from Yangjiang, Guangdong, bachelor's degree, working in China Southern Power Grid Digital Grid Group Information and Communication Technology Co., Ltd., engineer, research direction: cyberspace security.

Hong Zou (August 1986), male, Han nationality, born in Loudi, Hunan Province, master's degree, working in China Southern Power Grid Digital Grid Group Information and Communication Technology Co., Ltd, Senior engineer, research direction: network security, data security.

Zifeng Zeng (May 1997), male, Han nationality, from Yangjiang, Guangdong, bachelor's degree, working in China Southern Power Grid Digital Grid Group Information and Communication Technology Co., Ltd., engineer, research direction: cyberspace security.

Weijie Xu (July 1993) male, Han nationality, from Quanzhou, Fujian, bachelor's degree, working in China Southern Power Grid Digital Grid Group Information and Communication Technology Co., Ltd. research direction: cyberspace security.

Jiawei Jiang (August 1997) male, Han nationality, Fujian Jianouren, master's degree, working in China Southern Power Grid Digital Grid Group Information and Communication Technology Co., Ltd., title, research direction: cyberspace security.