Research on Network Security Vulnerability Detection Method Based on Artificial Intelligence

Guoyu Luo1, *

1School of Computer Science, Hubei University of Technology, Wuhan, 430068, China

*Corresponding Author’s E-mail: 877189650@qq.com

Abstract. The Internet is gradually being used in various fields, and people's lives are increasingly dependent on Internet technology. In network construction, we need to focus on maintaining network security and protecting the legitimate interests of users. In the network security defense system, network security vulnerability detection is an important part. Network attacks and network security vulnerability detection technology updates have shown a confrontational spiral in the field of network information security. One of the key contents of network security protection is how to effectively detect unknown network security vulnerabilities. In terms of network security vulnerability analysis, this paper studies cross-site scripting vulnerability analysis, SSL security vulnerability analysis, and binary vulnerability analysis; in terms of network security vulnerability prediction, it discusses random forest prediction models, convolutional neural network prediction models, and support vector machine predictions. Model; combined with common rules for network security vulnerability detection, comparative analysis of the network security vulnerability detection effects of several typical machine learning methods.

1. Introduction

With the continuous innovation and breakthrough of the new generation of information technology, the number of people using the Internet has grown rapidly. During the development of the Internet, various types of network security problems need to be resolved due to various forms of network attacks. In terms of network security vulnerabilities, the number of security vulnerabilities in various network service systems is increasing. Among them, vulnerabilities that can be exploited for remote attacks and high-risk vulnerabilities are more common. Common network intrusions include cross-site scripting, denial of service attacks, and SQL injection attacks. Due to the openness and complex topology of the Internet, the network traffic is relatively large and highly variable. These conditions have exacerbated the security vulnerabilities of network protocols, and many security vulnerabilities have been generated after the application of new technologies.

Network security vulnerability detection is an important research field of network information security. Taking SSL security vulnerabilities as an example, when a developer uses a self-signed certificate to establish an SSL connection, the server host name and authentication certificate need to be verified. In actual operation, if the relevant regulations are not strictly followed, SSL network security vulnerabilities will be brought. Among the network security vulnerabilities, SSL security vulnerabilities are mainly divided into four situations: calling insecure SSL-related interfaces, ignoring SSL errors, trusting all host names, and trusting all certificates.
There are large-scale outbreaks of network security vulnerabilities, and some major vulnerabilities are exposed every once in a while. Many cybersecurity incidents have formed some sabotage organizations, forming a black industrial chain of a certain scale. Faced with a large number of attack vectors, conventional methods such as manual analysis or rule engines cannot meet the needs of network security. Network attacks, the degree of damage, and the form of attacks continue to increase, and conventional network security vulnerability detection methods cannot keep up with the pace of attacks. Combining artificial intelligence with network security vulnerability detection and exploring automated network security solutions has gradually become an inevitable trend of development.

2. Network security vulnerability analysis

2.1. Cross-site scripting vulnerability analysis

Among the attacks against network system vulnerabilities, cross-site scripting (XSS) is a typical vulnerability that needs to be effectively dealt with. Most network vulnerability attacks involve only the attacker and the victim, while cross-site scripting vulnerability attacks involve the attacker, client and website.

2.1.1. Stored cross-site scripting vulnerability. Unlike reflected XSS attacks, attackers use XSS vulnerabilities to store malicious data in the database. When the server receives a request, it also reads malicious scripts when reading data from the database. When the data is displayed on a fixed page, the stored XSS attack will affect all users who visit the page. Stored XSS writes attack sentences in the database, which is severely destructive and has a wide range of harms.

2.1.2. Reflected cross-site scripting vulnerability. Reflected cross-site scripting vulnerabilities are widespread and easy to use. When web applications use dynamic pages to pass parameters, there is a risk of reflection vulnerabilities. This kind of page uses parameters to carry message text data, and when the page is loaded, the text content is fed back to the client. Reflective XSS is mainly presented in two ways based on page input and URL parameters. For page-based input, in the page containing the input area, first close and redefine the attribute tag of the input area, or splice or comment the code between two input areas, etc. Embed the JavaScript code block on the requested URL link to reflect malicious code to the requesting client. The data is requested from the browser, and the malicious code is reflected to the browser through the backend. The attack process is shown in Figure 1.

![Attack process diagram](image-url)

**Figure 1. Reflected XSS attack process**
2.2. SSL security vulnerability analysis
In a normal SSL protocol session, there is generally no man-in-the-middle attack when the SSL protocol handshake is used to verify the identities of the communicating parties. Due to the existence of SSL security vulnerabilities in some applications, attackers forge certificates and conduct man-in-the-middle attacks between the client and the server. The "man in the middle" is placed on the network communication link between the client and the HTTPS server, and establishes connections with the client and server respectively. In the process of server authentication, it intercepts and forges the server certificate and sends it to the client. Due to the existence of SSL security loopholes in the client, there is a lack of effective verification of forged certificates, and the "middleman" is regarded as the server. The "middleman" obtains confidential information such as negotiation keys and encrypted messages, and even tampered with transmitted data.

2.3. Binary vulnerability analysis
Binary code is the ultimate manifestation of the network system, and the network system contains binary execution programs. Due to the low readability and analysis of the binary code, the binary vulnerability analysis technology is based on the reverse analysis of the disassembly, and the binary bytecode is reversed into the assembly code and analyzed in depth. Compared with source code, binary code lacks program type and structure information, making analysis difficult. It is difficult to determine common vulnerability modes in binary code, it is difficult to accurately judge the boundaries of the buffer on the stack, it is impossible to parse instructions in the process of binary code upgrade, and even structural errors occur. Therefore, compared with source code analysis, the accuracy of subsequent analysis is lower. At this stage, there is a lack of mature automated analysis tools for binary vulnerabilities.

Artificial intelligence technology is developing rapidly. In order to better analyze the internal relationship and semantic information between programs, artificial intelligence-based network security vulnerability detection technology has become a research hotspot. Artificial intelligence-based vulnerability mining technology uses machine learning algorithms to train classification models through program-related attribute features to achieve classification. After the machine learning model is trained, it can process large-scale data with fast detection speed and low detection cost. Because it is difficult to extract effective features from binary files, the accuracy of network security vulnerability detection is low. Therefore, the binary vulnerability detection technology based on machine learning needs further research.

3. Network security vulnerability detection model
In the final analysis, network security vulnerabilities are code vulnerabilities. For code vulnerabilities, there are many related definitions in academia and industry, and they have different focuses under different application scenarios and premises. In computer network security, the problems that need to be solved are software vulnerabilties, vulnerabilities and defects. Network security vulnerability detection is, in the final analysis, a process of inferring whether there are vulnerabilities in code fragments through code features. We must first transform each module or fragment of the source code, obtain relevant features, use the data set representing the vulnerability feature, and obtain a model that can effectively identify the vulnerability through certain rule training, learning, and evaluation of the judgment model.

3.1. Random forest detection model
Multiple models are constructed using randomly selected attributes and integrated learning. It is mainly composed of three parts.

Sample set selection. On the original training set, there is self-sampling of replacement. After M random sampling, M sample data sets are obtained. After N rounds of sampling, N new training sets are obtained.

Generate a decision tree. Assuming that each training set has D features, in each process of generating a decision tree, K features are randomly selected from the current D features in the training set, and then
an optimal attribute is selected from the K features for division. Among them, the K value determines the degree of randomness. After N rounds of training, N independent decision tree models are obtained.

Model integration. In classification problems, the final output is the category that accounts for the majority of all decision tree model results; for regression problems, the results are obtained based on the average of all decision trees.

3.2. Convolutional neural network detection model
Convolutional neural network is suitable for processing data with spatial structure relationship, and it has made major breakthroughs in image classification, detection and segmentation tasks. Convolutional neural networks are good at learning the structural features of input data and are suitable for classification tasks. The convolutional neural network model structure is shown in Figure 2.

![Convolutional neural network model structure](image)

Figure 2. Convolutional neural network model structure

The embedding layer is used to process the data, and the vector representation of the data is obtained and then input to the one-dimensional convolutional layer to extract the feature map, and then perform the global maximum pooling on the feature map to obtain the feature corresponding to the convolution kernel. Combine the features through the fully connected layer and output the classification results. To build a deep convolutional neural network model, you need to select hyperparameters such as the size and size of the convolution kernel, pooling strategy, and activation function. The output dimension of the last fully connected layer of the deep convolutional neural network model is 1, and the output result of the activation function is used to detect whether the network data has loopholes.

3.3. Support vector machine detection model
Support vector machines have excellent performance in classification tasks. Support vector machines and kernel methods eventually become the basic technology of machine learning. It is to classify the samples by finding the hyperplane that can divide the sample to the maximum. For linearly separable data, we can directly build a support vector machine model; for linearly inseparable data, we can construct a kernel function to build a support vector machine model in a high-dimensional space.

Basic type is shown in formula (1) as follow:

\[
\min_{w,b} \frac{1}{2} \|W\|^2
\]

\[s.t. \ y_i(w^TX_i + b) \geq 1, \ i = 1,2,\cdots,m.\]  

Through Lagrangian multiplier method and KKT condition, the dual type is obtained, where \(K(x_i,x_j)\) is the kernel function identifier, as shown in formula 2.

\[
\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j k(x_i,x_j)
\]

\[s.t. \ \sum_{i=1}^{m} \alpha_i y_i = 0, \ \alpha_i \geq 0, \ i = 1,2,\cdots,m\]  

Support vector machines are usually solved using convex optimization techniques. SMO (Sequential Minimal Optimization) obtains the final support vector machine classification model by selecting a pair of variables that need to be updated and solving it iteratively. It is an efficient solution algorithm. The support vector machine only needs to use a part of the support vector data to construct a hyperplane decision surface, which is combined with the kernel function to solve various non-linear classification problems by selecting different kernels, and its classification accuracy is high.
4. Implementation and performance analysis of vulnerability detection model

In order to verify the effectiveness of the network security vulnerability detection model, this paper uses collected data and simulated experiments to predict the vulnerabilities in the binary code of the network system.

4.1. Model training

During the training process, this article uses Google's deep learning library TensorFlow and deep learning top-level library Keras as tools for building deep learning vulnerability detection models. Including basic vector matrix calculations, various optimization algorithms, the realization of basic units of various convolutional neural networks and recurrent neural networks, and the visual auxiliary tool Tensorboard. Keras is an independently developed deep learning library, which encapsulates commonly used neural network layers, simplifies the model building and training process, and lowers the threshold of deep learning. Keras uses simple and fast prototyping. It is highly modular, minimalist, and expandable. It supports CNN and RNN, or a combination of the two. At each training step, 32 small batches of sampled data are randomly selected from the data set to update the model, thereby stabilizing the training process. Since network security vulnerability detection is a binary classification problem, this paper uses binomial cross entropy loss as the objective function.

\[ L(Y, P(Y|X)) = -log P(Y|X) \]  \hspace{1cm} (3)

4.2. Test results and analysis

The prediction accuracy curve during the training of different models is shown in Figure 3, which includes four deep learning models and a traditional multi-layer perceptron model. In the first iteration, the accuracy curve of the deep learning model rose rapidly. After training once on the training set, the prediction accuracy can reach over 85%. In subsequent iterations, the accuracy curve rose slowly and remained basically unchanged after the second iteration. In the end, the average prediction accuracy of the three deep learning models in this article reached about 90%.

![Figure 3. Prediction accuracy curve during training of different models](image)

The loss function value curve in the training process of different models is shown in Figure 4. The loss function values of all deep learning models dropped significantly in the first iteration, reaching between 0.2-0.3. In the subsequent training process, the loss function value curve slowly drops and gradually converges, and finally stabilizes at about 0.18.
After 5 iterations, the prediction accuracy of the SVM model finally reached 71.25%. Compared with the deep learning network security vulnerability detection model, the prediction accuracy of the multilayer perceptron model is lower, and its loss function value is higher than that of the deep learning model. The experimental results show that the network security vulnerability monitoring method using artificial intelligence can obtain a better vulnerability detection effect. In the network security vulnerability detection method model based on artificial intelligence, the detection results of the deep learning model have a certain similarity, and the vulnerability detection effect of the deep learning model is better than that of the machine learning model.

5. Conclusion

In the network security vulnerability detection method based on artificial intelligence, random forest model, support vector machine model and deep learning model are all more effective methods. Through the research of traditional network security vulnerability detection methods and modern detection methods, after analysis and verification, deep learning network security vulnerability detection models can be obtained better vulnerability detection effect. Compared with traditional multi-layer perception models, deep learning models can further improve the accuracy of vulnerability detection. In further research, it is necessary to repeatedly train the model to compare the true rate, false positive rate and F value of different models on the test set. The F value of the convolutional neural network model is the highest. Analyzing the vulnerability detection accuracy and loss function value curve, the deep learning model based on convolutional neural network has fewer parameters, and can obtain faster training speed compared with the long-short-term memory network deep learning model.

References

[1] Sun L, Xu C, Hu X L. (2016) A prediction method of long non-coding RNA based on random forest. Journal of Yangzhou University (Natural Science Edition), 19(04): 50-53.
[2] Xie X L, Ye X D, Dong Y M. (2018) Gradient Boosting Random Forest Model and Its Application in the Forecast of Day-ahead Clearing Electricity Price. Computer Applications and Software, 35(09): 327-333.
[3] Zhao J Y, Lu H G, Jiang J P, et al. (2018) Research on fault diagnosis method based on improved decision tree. Journal of Chengdu University of Information Technology, 33(06): 624-631.
[4] Zhang C, Qiu T. (2019) Gas station sales forecast based on decision tree integration model. Computers and Applied Chemistry, 36(06): 615-619.
[5] Zhao L. (2018) Cloud computing resource load prediction model based on support vector machines. Journal of Nanjing University of Science and Technology, 42(06): 687-692.
[6] Wang J L, Tong E D, N W J, et al. (2018) Intrusion detection model based on CNN-NSVM. Information and Communication Technology, 12(06): 48-55.
[7] Zhao Z M, Yue Y Z. (2017) Security vulnerability detection simulation in the process of network information interaction. Computer Simulation, 34(11): 426-429.
[8] Lai Q, Zeng H W. (2017) Simulation research on rapid defense against attacks in network information transmission[J]. Computer Simulation, 34(11): 312-315.

[9] Zhang Z H. (2018) Real-time detection technology for network security vulnerabilities based on penetration testing. Science Technology and Engineering, 18(20): 297-302.

[10] Qi L J. (2019) Mobile network hybrid security vulnerability detection simulation under static defense. Computer Simulation, 36(04): 282-285.