Real-Time Network Intrusion Detection System Based on Deep Learning

Yuansheng Dong
Huazhong University of Science & Technology
Luoyu Road 1037, Wuhan, China
ysdhust@hotmail.com

Rong Wang and Juan He
Digital Society & Blockchain Laboratory
Beihang University, Beijing, P. R. China
{wangrong & xiongbao_hj }@buaa.edu.cn

Abstract—**Computer network is vulnerable to hackers, computer viruses, and other malicious attacks. As an active defense technology, intrusion detection plays an important role in the field of network security. Traditional intrusion detection technologies face problems such as low accuracy, low detection efficiency, high false positive rate, and inability to cope with new types of intrusions. To solve these problems, we propose a real-time network intrusion detection system based on deep learning, which uses big data technology, natural language processing technology and deep learning technology. Our main contributions are as follows:**

(1) Use Flume as the agent for log collection to realize real-time massive log collection, using Flink as real-time computation engine.

(2) Aiming at the high dimensional problem of traffic data, a self-encoder-based intrusion detection dimension reduction method is proposed, and the intrusion detection data is preprocessed, including data cleaning, coding, extraction and integration, and normalization.

(3) Propos a deep learning-based intrusion detection model, AE-AlexNet, which uses Auto-Encoder AlexNet neural network. The experimental results of the intrusion detection data set KDD 99 show that the accuracy of the AE-AlexNet, model is as high as 94.32%.

Keywords- intrusion detection; deep learning; self-encoder; adaptive acquisition

I. INTRODUCTION

Intrusion detection technology is an important guarantee for computer network security systems. which has been paid much attention by researchers in the field of network information security. Intrusion detection ensures the normal operation of computer network by collecting and analyzing various information and data on the network and identifying abnormal data. Intrusion detection system (IDS) is a "burglar alarm" in the field of computer security. Its purpose is to defend the system by combining the alarm issued when the network security is threatened with the inspection entity. Intrusion detection is a good complement to firewall, which improves the system's ability to deal with network attacks, reduces economic losses for enterprises, and provides users with better services.

Early intrusion detection research was based on expert knowledge, Dorothy Denning and Peter Neumann co-developed the Intrusion Detection Expert System (IDES) [1]. And the behavior that causes the network system to be in an illegal state is called the intrusion behavior [2], which indicates that the intrusion detection research has arisen. In 1986, Neumann and Denning proposed the real-time IDS model in the world that mainly studies the user behavior profile.

In 1987, Denning proposed a research based on the research of the intrusion detection expert system. The abstract model of IDS [3], which is a new approach to computer security, differs from traditional encryption and access control technologies. In 1990, Heberlein proposed Network Security Monitor (NSM) [4]. Since then, a lot of later IDS research is based on The NSM system architecture is based on basic research. Therefore, NSM is an important milestone in the history of IDS systems, network-based IDS (NIDS) and host-based of IDS (HIDS) were formally formed.

Nadeem [5] combines neural network with semi-supervised learning, and uses a small number of labeled samples to obtain high accuracy. Staudemeyer [6] implemented intrusion detection based on LSTM regression neural network. The results show that the LSTM classifier has certain advantages over other strong static classifiers. These advantages lie in detecting DoS and Robe attacks, both of which can produce unique time series characteristics. In order to compensate for the high false alarm rate, Kim [7] proposed a call language modeling method to improve the host intrusion detection system based on LSTM. Agarap [8] replaced Softmax by introducing linear support vector machine (SVM) into the final output layer of GRU model, and applied the model to the second classification of intrusion detection. Kolosnjai [9] realized a neural network consisting of convolution and feedforward neural structures. This paper presents a hierarchical feature extraction method, which vectorizes n-gram instruction features and convolution features.

Existing problems of IDES:

(1) The recognition rate is not high enough. The existing intrusion detection technology has a low accuracy rate and a high false positive rate and a false negative rate. With the increasing amount of network data, the diversification of intrusion behavior and the better disguise of intruders had become the bottleneck of intrusion detection.

(2) Poor adaptive ability. The adaptive ability mainly refers to the detection ability of the new intrusion behavior, and the existing intrusion detection technology always has certain defects in this aspect.

978-1-7281-0945-9/19/$31.00©2019 IEEE

(3) The efficiency of processing large amounts of data is too low. With the advent of the era of big data, a large amount of network data will be generated in a short period of time. Intrusion detection is to identify intrusion behaviors in these large quantities of data, which brings great challenges to current intrusion detection technologies.

In order to solve the problems mentioned above, we propose a real-time network intrusion detection system based on deep learning, which uses big data and deep learning technology to achieve real-time network intrusion detection, content of the paper is organized as follows: Section 2 presents related technologies background; section 3 presents the overall system architecture, and system function design; section 4 of the proposed test analysis system; section 5 of the work of this paper is conclusion and future works.

II. Relation Works

A. Intrusion Detection

The intrusion detection system is mainly divided into three parts, data collection module, data analysis module and intrusion response module. The data analysis module is the core of the intrusion detection system, and the intrusion detection technology is the core of the data analysis module. Intrusion detection technology is mainly divided into two types of anomaly detection technology and misuse detection technology [10].

(1) Anomaly detection technology

Anomaly detection principle is the belief that any normal behavior patterns are there is a pattern, by analyzing the system log files and audit user behavior and other channels to collect relevant data, sorting out the normal user's profile, the current The data is compared to the normal usage pattern, and if the deviation is too large, it is considered to have occurred. On the basis of constantly updating the behavior patterns of normal users, selecting appropriate thresholds for judgment can obtain better detection accuracy. Anomaly detection techniques also use many mathematical or statistical models. There are mainly anomaly detection methods based on probabilistic statistical models, anomaly detection methods based on cluster analysis [11], anomaly detection methods based on neural networks [12], anomaly detection methods based on data mining [13] and so on. These methods have been widely used in intrusion detection, but these technologies have their own advantages and disadvantages, and different technologies need to be selected according to the actual situation.

(2) misuse detection

Misuse detection technology is mainly to establish an accurate intrusion feature library to describe the characteristics of intrusion behavior, intrusion conditions, etc. to effectively detect intrusion behavior. It can be seen that the focus of misuse detection is to summarize the intrusion behavior signature database. The principle of misuse detection is relatively simple, scalable, high detection efficiency, and good real-time performance. However, misuse detection also has certain defects. Misuse detection technology can only maintain high detection accuracy for known types of attacks. However, the detection efficiency of unknown attacks will be greatly reduced, and the workload of maintenance and update of the intrusion signature database is very large, and the portability is not strong and does not have universality. Commonly used misuse detection methods are: misuse detection method based on expert system [14], misuse detection method based on pattern matching [15] , misuse detection method based on state transition [16] , error based on information feedback Using detection methods.

B. Deep Learning Technology

Deep learning further assumes that the process of this interaction can be divided into multiple levels, representing a multi-layered abstraction of observations. Different layers and layers can be used for different levels of abstraction. Deep learning uses this hierarchical abstraction. Higher-level concepts are learned from low-level concepts. Deep learning can automatically extract high-dimensional features of data and discover the correlation between data.

This model was put forward by Alex [16], and won the 2012 ImageNet championship. It refreshed the record of image classification and determined the position of Deep Learning in computational vision at one stroke. And AlexNet has attracted wide attention.

C. Big Data Technology

Flume is a highly available, highly reliable, distributed, massive log collection, aggregation and transmission system from Cloudera. Flume supports the customization of various data senders in the log system for data collection. At the same time, Flume provides the right the data is simply processed and written to the ability of various data recipients (customizable). Use Flume to collect various data resources such as web logs, system logs, application logs, events, etc., and then combine with distributed file systems HDFS centrally stores these huge amounts of data, providing the most detailed log data for intrusion detection.

Flink is an open source stream processing framework developed by the Apache Software Foundation. Its core is a distributed streaming data stream engine written in Java and Scala. Flink executes arbitrary stream data programs in a data-parallel and pipelined manner, and Flink 's pipeline runtime system can execute batch and stream handlers. In addition, Flink's runtime itself supports the execution of iterative algorithms. The Flink program is mapped to the stream data stream after execution, each Flink data stream starting with one or more sources (data input, such as a message queue or file system) and one or more receivers (data output, such as messages) End of queue, file system or database, etc.). Flink convection can perform any number of transformations, these streams may be organized as a directed acyclic dataflow graph, and allows the program branches to be combined with the data stream.

III. Real-Time Network Intrusion Detection System Based on Deep Learning

A. System Architecture

The overall architecture design of real-time network intrusion detection system based on deep learning is shown in Figure 1. The bottom layer is the data acquisition layer, which provides the ability to collect vast amounts of data from the enterprise. The data storage layer is responsible for the distributed storage of massive data and the ability of Extract-Transform-Load (ETL). Data analysis and calculation layers provide flow computing power, data analysis, feature extraction and calculation functions, wherein the depth of the neural network model used AlexNet. Data presentation layer provides intrusion detection, warning and safety features such as security analysis, by the self-analysis tool JAVA WEB provides capabilities.

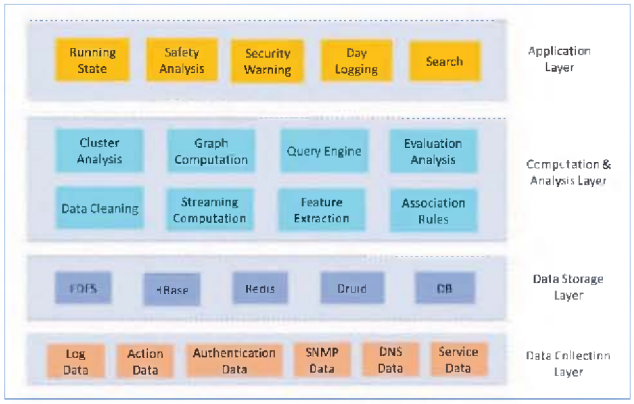

Figure 1 System achitecture

B. Data Collection Module Design

The log system can monitor incoming audit data sources. Information systems in different industries are generally different, different hardware, different operating systems, and different applications. Therefore, different types of data sources need to be processed, and various data sources have different security levels. For example: router logs, firewall log information, and network management logs.

The underlying data processing of the real-time log analysis system is divided into three parts: acquisition, cleaning, and indicator calculation, as shown in Figure 2. The acquisition module collects the logs of each data source, and send Kafka in real time through Flume. The cleaning module receives log data in real time, perform data processing and conversion, and the cleaning task is based on Flank. Currently, one billion-level traffic data is processed every day. The structured data after cleaning task will be sent to Kafka queue index calculation again. Real-time receiving structured traffic data from Kafka, real-time calculating related indicators, index calculation task is based on Flink. Tasks have the advantages of low latency, high throughput, support for standard SQL, simple development, exactly-once semantics, support for window function calculation, etc. After index calculation, the data are mainly stored in HBase, Druid and other storage engines. The business system reads the real-time calculated index data and provides data analysis services for operators.

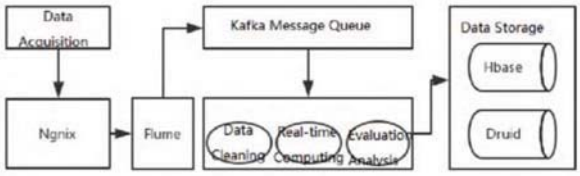

Figure 2 Data collection flow

C. Design of intrusion detection module

The intrusion detection model is designed as shown in Figure 3, which is mainly composed of three parts: data preprocessing, dimension reduction of Auto-Encoder (AE) features, and data classification after dimension reduction of AE.

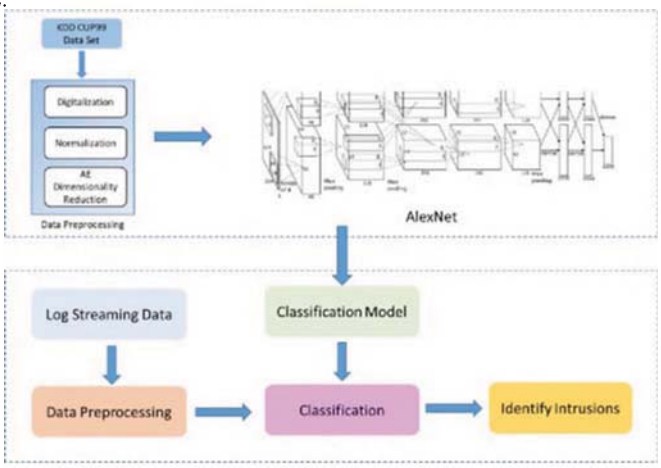

Figure 3 Intrusion detection module

(1) Data preprocessing

There are three main functions of data preprocessing: one is to transform the original data which can not be directly used as input into data that conforms to the model input format. Secondly, the accuracy of the model can be improved by normalizing the input data and thermal coding.

Using KDD 99 data set as experimental data, which is an intrusion detection data set with recognized standards and excellent performance. In order to make the dimension of data comparable and eliminate the influence of dimension, all data records in training set and test set should be normalized.

(2) Self-encoder feature dimension reduction

Auto-Encoder Network (AE Network) is an effective nonlinear dimensionality reduction method proposed by Hinton [19]. It uses multiple hidden neural network layers to nonlinearly transform the input high-dimensional data sets, and the original high in the process of unsupervised learning. Dimensional features are mapped into low-dimensional features, and these low-dimensional features are reconstructed into high-dimensional features. In this way, the feature dimensions of the data set are effectively reduced to the extent

that the feature information is guaranteed to the maximum extent.

(3) AE- AlexNet classifier for classification

AlexNet can be divided into multiple levels, and the different layers and layers can be used for different levels of abstraction. We use the KDD 99 network intrusion data set as the training set data, obtain the original data according to the data acquisition module, and process the data, then use the AE-AlexNet classifier to classify and judge the result of the test data to achieve the purpose of intrusion detection.

IV. Experimental Simulation

The data set used in this experiment is the KDD 99 network intrusion data set, which consists of 5 million records. It also provides 10% training data subset and test data subset. This experiment randomly selected from a training data set of 137,200 data as a training set of data, randomly picked from a subset of the test strip 2000 taken subjected to testing. The test results are listed in Table 1 below. The experimental results show that the AE-AlexNet model has a high detection rate for attacks in the current environment, and the total detection rate is 94.32%.

Table 1 Experimental simulation results

Attack Type	Samples Number	Detection Rate	Total Detection Rate
Normal	1056	100	
DoS	120	95.24	
Probe	426	95.75	
U2R	91	89.76	94.32%
R2L	54	92.78	
Vulnerability	253	92.39	

V. Conclusion and Future Work

This paper presents a real-time network intrusion detection system based on deep learning, which through the system Flume collects log information and network information, uses Flink to perform real-time cleaning and feature extraction on the original data, and then transmits the extracted high-order features to the neural network for training and judgment. When the event reaches the alert value, the threat is identified, a decision is made, or an alert message is sent. Experiments show that the training of neural networks can improve the accuracy of identifying threats. The learning of neural networks has obvious advantages in identifying threats, making neural network technology possible in intrusion detection systems. At the same time, there are some shortcomings in the actual test, mainly in the long training period and poor portability. In the future, we will improve the performance and function of the system.

References

[1] Lunt, Teresa F., and R. Jagannathan. "A prototype real-time intrusion-detection expert system." Proceedings. 1988 IEEE Symposium on Security and Privacy. IEEE, 1988.

[2] Anderson, James P. "Computer security threat monitoring and surveillance." Technical Report, James P. Anderson Company (1980).

[3] Denning, Dorothy E. "An intrusion-detection model." IEEE Transactions on software engineering 2 (1987): 222-232.

[4] Heberlein, L. Todd, et al. "A network security monitor." Proceedings. 1990 IEEE Computer Society Symposium on Research in Security and Privacy. IEEE, 1990.

[5] Nadeem, Mutahir, et al. "Semi-supervised deep neural network for network intrusion detection." (2016).

[6] Staudemeyer, Ralf C. "Applying long short-term memory recurrent neural networks to intrusion detection." South African Computer Journal 56.1 (2015): 136-154.

[7] Kim, Gyuwan, et al. "LSTM-based system-call language modeling and robust ensemble method for designing host-based intrusion detection systems." arXiv preprint arXiv:1611.01726 (2016).

[8] Agarap, Abien Fred M. "A neural network architecture combining gated recurrent unit (GRU) and support vector machine (SVM) for intrusion detection in network traffic data." Proceedings of the 2018 10th International Conference on Machine Learning and Computing. ACM, 2018.

[9] Kolosnjaji, Bojan, et al. "Deep learning for classification of malware system call sequences." Australasian Joint Conference on Artificial Intelligence. Springer, Cham, 2016.

[10] Butun, Ismail, Salvatore D. Morgera, and Ravi Sankar. "A survey of intrusion detection systems in wireless sensor networks." IEEE communications surveys & tutorials 16.1 (2013): 266-282.

[11] Lankewicz, Linda Bright. "A nonparametric pattern recognition approach to anomaly detection." (1992).

[12] Shun, Jimmy, and Heidar A. Malki. "Network intrusion detection system using neural networks." 2008 Fourth International Conference on Natural Computation. Vol. 5. IEEE, 2008.

[13] Sequeira, Karlton, and Mohammed Zaki. "ADMIT: anomaly-based data mining for intrusions." Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2002.

[14] Porras, Phillip A., and Peter G. Neumann. "EMERALD: Event monitoring enabling response to anomalous live disturbances." Proceedings of the 20th national information systems security conference. 1997.

[15] Kumar, Sandeep. Classification and detection of computer intrusions. Diss. PhD thesis, Purdue University, 1995.

[16] Ilgun, Koral, Richard A. Kemmerer, and Phillip A. Porras. "State transition analysis: A rule-based intrusion detection approach." IEEE transactions on software engineering 3 (1995): 181-199.

[17] Iandola, Forrest N., et al. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size." arXiv preprint arXiv:1602.07360 (2016).

[18] Olusola, Adetunmbi A., Adeola S. Oladele, and Daramola O. Abosede. "Analysis of KDD'99 intrusion detection dataset for selection of relevance features." Proceedings of the World Congress on Engineering and Computer Science. Vol. 1. WCECS, 2010.

[19] Hinton, Geoffrey E., and Ruslan R. Salakhutdinov. "Reducing the dimensionality of data with neural networks." science 313.5786 (2006): 504-507.