Web Log Anomaly Detection Based on Isolated Forest Algorithm

Wei Zhang
School of Computer Science & Technology
Xi'an University of Posts & Telecommunications
Xi'an ,China
zw_luciano@163.com

Lijun Chen
School of Computer Science & Technology
Xi'an University of Posts & Telecommunications
Xi'an ,China
cljcore@126.com

Abstract—**In order to improve the detection rate and time efficiency of the web log based anomaly detection method, a web log anomaly detection method based on the isolated forest algorithm is proposed. By analyzing the web log anomalies, an anomaly detection framework based on web logs was constructed, the web features for anomaly detection are extracted, a mathematical model is built for the web log and the feature values are calculated, and an anomaly detection is performed using the isolated forest. Compared with other methods through experiments, the results show that the method has greatly improved detection accuracy and detection efficiency.**

Index Terms—**Isolated forest, web log, log mining, anomaly detection**

I. Introduction

With the development of modern Internet technology, various web-based applications play an important role in human life. While bringing convenience to human beings, the Internet is also suffering from various attacks from all over the world. Anomaly detection is the most important part of computer security protection. The anomaly detection system can monitor the hosts and applications in the network in an all-round way, and can identify abnormal behaviors, effectively making up for the shortcomings of traditional security protection technologies, Maximize the security of computer systems and reduce the harm that security threats pose to the system. Therefore, the research on anomaly detection is a hot research topic in the field of computer security.

Traditional log analysis mainly involves artificially formulating various rules to match known anomalies. This method is effective for detecting anomalies when the log data is small and the type of exception is known. But in the era of big data, it is very difficult to manually analyze logs, and it is more difficult to obtain exceptions through manual analysis.

At present, in the aspect of log-based anomaly detection, Gao Yun et al.[1] proposed a log anomaly detection algorithm CADM based on grammar compression, which uses the relative entropy between the normal log and the log to be detected as an indication of the degree of abnormality. Wang Zhiyuan et al[2]. proposed an anomaly detection technology based on log template. First, text clustering based on editing distance to form a log template. Based on this, a feature vector is constructed, and a weak classifier is used to train to form a scoring feature vector. A strong classifier is constructed using the score feature vector and the random forest to determine whether an abnormality has occurred. Liu Zhihong et al. [3]used log analysis technology to analyze the process of massive Web access logs, and used the methods of feature character matching and access frequency statistical analysis to mine attacks. Vinayalumar et al.[4]proposed an anomaly detection algorithm for log sequence modeling using LSTM neural network.

Most of the above research work is based on statistical and pattern matching from some abnormal features for direct detection, lack of in-depth analysis of log behavior characteristics, Moreover, the detection algorithm used is inefficient, and it is difficult to effectively detect abnormal behavior in the face of an unknown web exception. In view of the above shortcomings, this paper proposes to use the isolated forest algorithm [5][6] to detect the abnormality of the web log, and build the anomaly detection model to perform data mining on the web log to realize the abnormal analysis of log. The experimental results prove that this method can improve the anomaly detection rate and also reduce the time consumption of anomaly detection.

II. Anomaly Detection And Web Log Analysis

A. Anomaly Detection Model

According to different detection methods, intrusion detection is mainly divided into misuse detection and anomaly detection. In the misuse detection method, the abnormal behavior is first defined, and then all other behaviors are defined as normal. By misuse detection, any unknown behavior is considered normal. Anomaly detection can discover abnormal behaviors in networks and systems by collecting and analyzing data generated by computer operations[7], and then judge whether it invades according to some decision algorithm. Compared with misuse detection, anomaly detection can detect unknown anomalies and attacks, and has a wide range of applications in computer network security defense.

The difficulty in performing anomaly detection is how to build an anomaly detection model. In the case of the same data set, different detection models have great differences in the effect of anomaly detection. The commonly used anomaly detection models are mainly divided into two categories: statistical-based hypothesis testing models and machine learning-based anomaly detection models. The statistical-based model first assumes that the data to be tested is subject to a certain distribution (such as a Gaussian distribution). This method is only applicable to one-dimensional data, and has a great limitation on the detection of high-dimensional data. The machine learning-based anomaly detection model can be divided into anomaly detection model based on data mining, anomaly detection model based on neural network and anomaly detection model based on decision tree. The data set of anomaly detection is often unlabeled, and the training data does not indicate which are abnormal points. Therefore, In the big data environment, anomaly detection methods based on unsupervised machine learning have been widely used. The

978-1-7281-2348-6/19/$31.00 ©2019 IEEE

machine learning based anomaly detection model is shown in Figure 1.

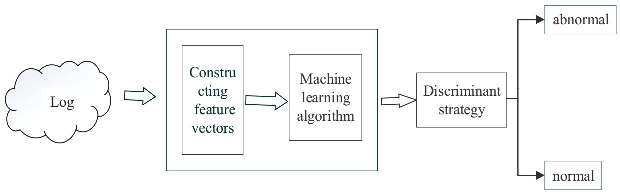

Fig. 1. Anomaly detection model based Machine learning

B. Log Analysis and Processing

The web log file records the specific situation of the user accessing the website. Each time the user initiates an access request, the server system generates a log record. The log information recorded by the server includes the requester address, the content of the request, the method of the request, the response status of the server, etc. The server records all relevant information of the web service through the log, so the log data recorded by the server can be used for data mining and anomaly detection[8]

Logs obtained from web servers have data omissions, data redundancy, and noise interference. Before using for machine learning analysis, data preprocessing is first required. The process of log data preprocessing mainly includes data purification, data formatting, user identification, session identification, etc. [9]. The data preprocessing process is shown in Figure 2.

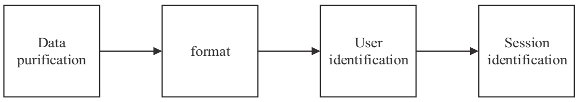

Fig. 2. Log preprocessing

Data cleansing is done to clean up redundant data and noise data.

The purpose of formatting is because the URL field information of the web log is incomplete, so it needs to be supplemented. Formatting the web log facilitates subsequent user identification and session identification.

Because there are proxy servers on the network, it is difficult to accurately identify each user in the web log[10], so the log records should be analyzed in detail to distinguish users in various situations.

The main purpose of session identification is to reconstruct the sequence of behaviors of user access from a large number of messy web logs[11].

There are various web attack methods. Some of the more common attack methods include injection attacks, command execution vulnerability attacks, cross-site scripting attacks, and directory traversal attacks. This article takes the common Nginx log format as the research object. Nginx uses the "main" log format by default. The default "main" log format and description of each field was shown in Table I:

TABLE I. NGINX LOG FIELD DESCRIPTION

Log field	*Field meaning*
$remote_addr	Client's ip address
$remote_user	Remote client user name
$time_local	Access time and time zone
$request	Requested URL and request method
$status	Response status code
$body_bytes_sent	The size of the file content sent
$http_user_agent	Agent used by the user
$http_x_forwarded_for	Record the client's ip address through a proxy server
$http_referer	Record user access links

By extracting the characteristics of each field, potential log associations can be mined and the web log behavior characteristics can be characterized to the greatest extent. The log feature information extracted mainly in this paper includes

(1) remote_addr:

(2)request:

This item contains more important information, including three parts:

Method: There are mainly GET/POST/HEADS and so on.

Resource: Display the URL of the resource

Protocol: Display protocol and version information, usually HTTP/1.1 or 1.0

The text feature data is first vectorized by n-gram, and then the text feature attribute replacement is performed using TF-IDF, this is a weighting technique commonly used in information retrieval and text data mining. After selecting the text features to be extracted, use n-gram to perform vectorization processing of feature data, and then use TF-IDF. Converting text feature attributes, The mathematical expression of TF-IDF is:

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,k}} \qquad (1)$$

Where $TF_{i,j}$ represents the frequency of occurrence of the keyword j in the document i. $n_{i,j}$ represents the number of times the keyword j appears in the document i.

The document frequency IDF indicates how often words appear in the entire file thesaurus,, and the IDF is the "reverse file frequency". Its calculation formula:

$$IDF_i = \log \frac{|D|}{|j: t_i \in d_i| + 1} \qquad (2)$$

IDF_i stands for the inverse document frequency of the word i. |D| indicates the total number of files in the document library. $|j: t_i \epsilon d_i|$ represents the total number of documents containing the word i.

The feature extraction of the web log is a method of using word segmentation to map the abnormal keywords in the web log into the space vector. The method regards the feature as a text. First, the word segmentation is performed. After the word segmentation, the space vector can be constructed. Each feature in the space vector is the existing feature keyword.

III. Web Log Anomaly Detection Based On Isolated Forest

A. *Isolated Forest Algorithm*

The algorithm of iForest is: first use a random hyperplane to cut the data space, and each time it is cut, two subspaces are generated. The random hyperplane is then used to cut each subspace, continuously cutting the data subspace until there is only one data point in each subspace. Since the data set with high density needs multiple cuts to stop, the point with low density will stay in a subspace very early after cutting, so the low density point that is cut out is the abnormal point that is isolated.

Given n sample data X = {x1, x2 ... xn}, the dimension of the feature is d, the feature q of one sample and the segmentation value p are randomly selected, and the data set X is recursively segmented. iForest consists of k iTrees (isolated trees). The method for building iForest is as follows:

1) Arbitrarily select k data in the training set as the root node of iForest.
2) Arbitrarily specify a dimension to randomly generate a cut point p in the current node data.
3) A hyperplane is generated at point p, and the current data space is divided into 2 subspaces using the hyperplane. Put the data smaller than point p in the left subtree of the current node, and put the data larger than p in the right subtree of the current node.
4) Recursively split sample space in a subtree, and repeat the construction of the new child node until any of these conditions are met: (1) the height of the isolated tree reaches the limit value (2) there is only one sample on the node.

B. *Outlier detection*

Any training data x is traversed by each iTree, and then x is calculated to fall on the first few layers of each tree. Finally, the average value of x in each tree is obtained. After each test data is traversed through iTree, the average path length of each data in iTree is calculated. A threshold can be set, and the test data below this threshold is abnormal. For a data set containing n samples, the average path length of iTree is:

$$c(n) = 2H(n-1) - \frac{2(n\text{-}1)}{n} \qquad (3)$$

Where $\mathrm{H(i)}$ is the harmonic function and the value can be estimated as $\ln(\mathrm{i}) + 0.5772156649$. $c(n)$ is The average of the path lengths for a given number of samples n , used to normalize the path length $\mathrm{h(x)}$ of the sample x. The abnormal value of sample x is calculated as:

$$S(x,n) = 2^{-\frac{E(\mathrm{h(x)})}{c(n)}} \qquad (4)$$

Where $E(h(x))$ is the mathematical expectation of the path length of the sample x in a batch of isolated trees, from which the following conclusions can be drawn:

1) When $E(h(x)) \to c(n), \mathrm{s} \to 0$ When the average length of the path of the sample x is close to the average path length of the tree, it is impossible to distinguish whether it is an abnormality.

2) When $E(h(x)) \to 0, \mathrm{s} \to 1$ When the abnormal score of x is close to 1, it is determined to be abnormal.

3) When $E(h(x)) \to n-1, \mathrm{s} \to 0$. It was judged to be normal.

C. *Anomaly Data Detection Based on Isolated Forest Algorithm*

The isolated forest algorithm is used to detect the abnormality of the web log. First, the collected raw data is cleaned, the dirty data is removed, the redundant data is deleted, the data features are extracted, and then the iTree is constructed. The implementation process is as follows:

1) Arbitrarily choice a log attribute feature as a detection indicator;
2) Arbitrarily selecting a value k of the feature;
3) Among the features, less than k is placed on the left, and greater than k is placed on the right;
4) The loop constructs the left and right branches, and stops constructing when the incoming data record does not change or the depth of the tree reaches the set value.

IV. Experiment Analysis

A. *Experiment Environment and Content*

CPU: 3.7GHz; memory: 16GB; operating system: Windows 10; development environment: python3.7

The experimental data used the public data set DARPA 99 log audit data. After preprocessing the data, 65536 valid web logs are obtained. The ratio of the training set to the test set is 8:2.

B. *Log Anomaly Detection Based on SVM*

Web log anomaly detection using SVM mainly includes two steps of data processing and model training: the data processing part first obtains the parameter feature value of the web log data by using the string matching algorithm, and then performs normalization processing on the data for the feature value. The model training uses libsvm to implement the SVM classification algorithm. By modifying the input of the function in libsvm, different kernel functions are used to compare the detection effects. The following table shows the changes in the detection rate when the Gaussian kernel is used. Where C is the function penalty parameter and σ is the bandwidth of the Gaussian kernel.

TABLE II. Detection Rate at Different Parameters

σ *C*	*1*	*2*	*4*	*10*	*50*
1	0.74	0.73	0.72	0.62	0.57
2	0.73	0.71	0.68	0.62	0.56
3	0.72	0.71	0.67	0.61	0.56
4	0.71	0.70	0.67	0.61	0.55

When C=1.4,σ=0.5, the detection rate of Gaussian kernel function is 75%, and the detection rate of SVM algorithm of each kernel function is as follows:

TABLE III. COMPARISON OF DIFFERENT KERNEL FUNCTION DETECTION RATE

Kernel Function	*Precision*	*Recall*	*F-Measure*
Polynomial kernel	71.6%	75.2%	72.4%
Sigmoid	70.4%	73.1%	72.4%
Laplace	71.2%	73.2%	72.6%
Gaussian kernel	72%	75%	73%

It can be seen from the table that SVM has a better detection effect when using the Gaussian kernel function.

C. Log Anomaly Detection Based on K-Means

The k-means algorithm selects the desired cluster center k. by continuously iterating and recalculating the cluster centers to minimize the variance within the cluster, compact and independent clusters are obtained as the ultimate goal of the algorithm. Using the function of the extremum method, adjust the iteration number threshold to get the best clustering effect.

Although the algorithm has the advantages of high efficiency and simplicity, it is very sensitive to abnormal point and noise point data. Even a small amount of abnormal data will have a great influence on the performance and results of the algorithm, which will cause the average value to deviate. And the clustering result of the algorithm has strong dependence on the initial value selection. Once the initial cluster center selection is not good, the best detection result may not be obtained.

Therefore, use the following method to improve the k-means to detect the abnormality of the log.

1) Before using the k-means algorithm, first obtain the abnormality coefficient of each data sample, mark all the samples whose abnormality coefficient value is greater than the threshold as abnormal outliers, and then filter the points, and do not calculate the cluster center. Use these anomalous data points and outlier data points.

2) When the cluster center is initialized, the random initialization method is not used. Instead, the range of values of the data samples except for the outliers and outliers in each dimension is first calculated, and the cluster number is calculated as needed, and the average difference is used. Initial clustering center.

Iterative algebra F=100 when clustering with K-means algorithm, Mins=100 for class density, w=1.45 for density threshold, and analysis when clustering values are k=2,k=3 and k=4. Use accuracy, error rate as an indicator of k-means detection effect, Accuracy is the number of abnormal records detected divided by the total number of abnormal records in the test set. The error rate is the number of normal records that were misjudged by the intrusion divided by the total number of normal records in the test set. The abnormal detection effect is as follows:

TABLE IV. ANALYSIS OF TEST RESULTS

K	*Accuracy*	*Error Rate*
2	62%	0.51%
3	75.6%	0.32%
4	83%	0.18%

D. Log Anomaly Detection Based on Isolated Forest

The number of initial isolated trees in the isolated forest algorithm is 100, and the number of random samples selected for each isolated tree is 256. As the number of iTrees and the total number of samples increase, the calculation time also increases. When the number of selected iTrees reaches a certain value, the anomaly detection rate increases. The number of isolated trees and the number of random samples are two key factors affecting the algorithm of isolated forests. The effect of using different isolated trees and random sample numbers on the detection rate is shown in Figure 3 and Figure 4.

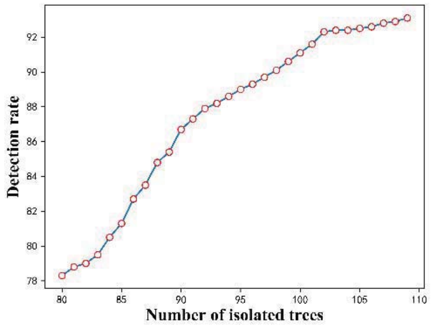

Fig. 3. The effect of isolated tree numbers on detection rate

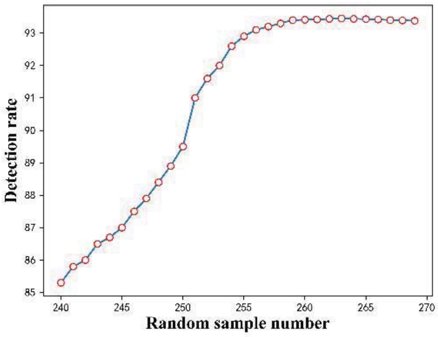

Fig. 4. The effect of random sample number on detection rate

It can be seen from Fig. 3 that the anomaly detection rate increases gradually with the number of isolated trees constructed. When the number of isolated trees reaches 105, the anomaly detection achieves a good effect, and then the number of isolated trees increases. The increase in detection rate is limited. In Fig. 4, The accuracy of anomaly detection increases gradually with the number of selected subsamples. When the number of sub-samples reaches 260, the accuracy rate achieves better results. When it exceeds 260, the accuracy improvement slows down. For this experiment, the isolated tree was selected as 105, and the random sample was 260 for abnormality detection.

E. Analysis of Results

This article uses accuracy and recall to measure the validity of experimental results. Among them, precision indicates the precision, recall indicates the recall rate, and f-measure is a compromise between the two indicators. Its mathematical definition is as follows:

$$recall = \frac{TP}{TP+FN} \tag{5}$$

$$precision = \frac{TP}{TP+FP} \tag{6}$$

$$f-measure = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{7}$$

In order to verify the detection rate of this method, k-means, SVM, and isolated forest algorithms were used to perform experiments on the same data set. The comparison of different algorithm detection results is shown in Figure 5.

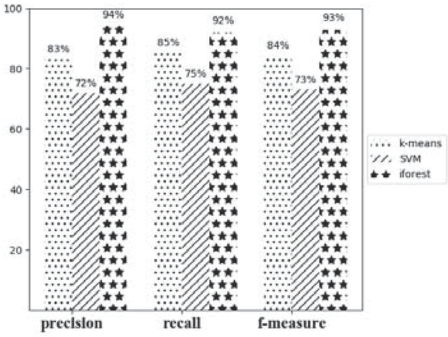

Fig. 5. Comparison of detection effects of different detection algorithms

By analyzing Figure 5, the precision of the isolated forest algorithm is 94%, the recall rate is 92%, and the f-measure is 93%. Under the same conditions, the iForest method has higher precision and recall rate than SVM and K-means methods.

In order to compare the time consumption of the isolated forest method with other methods, 1024 samples were initially selected for simulation, and then the sample iteration was doubled each time to calculate the time consumed. The time efficiency of the different methods is shown in TABLE IV.

TABLE V. TIME COMPARISON OF DIFFERENT METHODS

Methods *Number of Logs*	*Isolated Forest Method*	*K-Means*	*SVM*
1024	2.31	0.88	0.10
2048	2.28	2.12	0.68
4096	2.78	4.35	2.61
8129	3.63	7.63	9.59
16384	5.34	20.19	37.04
32768	9.57	40.20	152.82
65536	17.09	148.26	551.88

Analysis Table V finds that when the amount of data is small, the three methods consume less time, and the k-means method and the SVM method consume less time in a smaller amount of data. When the amount of data continues to double, the execution time of the isolated forest method is slower, compared to the other two methods, the execution time is multiplied.

In summary, Using the isolated forest algorithm to detect anomalies in web logs can significantly improve the detection rate. The algorithm has linear time complexity, so it has practical value in the application of anomaly detection.

V. CONCLUSION

In this paper, the isolated forest algorithm is used to detect the abnormality of the web log. The algorithm does not need to label the data during the anomaly detection, and has linear time complexity. It has good stability when dealing with multi-dimensional massive data. Compared with other machine learning algorithms, it has higher detection rate and faster time efficiency. Since the hyperplane selected when randomly cutting the number of sub-samples will affect the detection rate of the algorithm, the subsequent research work should be optimized on the selection of the cutting hyperplane to improve the detection rate of the algorithm.

REFERENCES

[1] Gao Yun, Zhou Wei, Han Jizhong, Meng Dan.A Log Anomaly Detection Algorithm Based on Grammar Compression[J].Chinese Journal of Computers,2014,37(01):73-86.

[2] Wang Zhiyuan, Ren Chongguang, Chen Wei et al. Anomaly Detection Technology Based on Log Template[J]. Intelligent computers and applications,2018,8(05):17-20+24.

[3] Liu Zhihong,Sun Changguo.Actual Behavior Detection Based on Web Access Log[J].Computer and Network,2015,41(13):62-64.

[4] Vinayakumar, R., Soman, K. P., & Poornachandran, P. (2017). Long short-term memory based operation log anomaly detection. In International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017 (pp. 236–242). IEEE.

[5] Liu, F.T., Kai Ming Ting, Zhi-Hua Zhou. Isolation Forest[P]. Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on,2008.

[6] Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. Isolation-based anomaly detection.ACM Transactions on Knowledge Discovery from Data (TKDD)6.1 (2012): 3.

[7] Zhu Jiajun, Chen Gong, Shi Yong et al. Detection of abnormal behavior based on user portraits[J].Communication Technology, 2017,50(10):2310-2315.

[8] Mei Xiaohui. Application of cluster-based outlier mining in intrusion detection [D]. Chongqing University, 2015.

[9] Lin Xu. Research on anomaly detection technology based on WEB access log [D]. Ocean University of China, 2015.

[10] Gao Yang. Research on Intrusion Detection Algorithm Based on WEB Log [D]. Beijing University of Posts and Telecommunications, 2018.

[11] Liu Kai. Design and implementation of anomaly detection system based on log feature[D].Xi'an University of Electronic Science and Technology, 2014.

[12] Jing Yu, Dan Tao, Zhaowen Lin.a hybrid web log based intrusion detection model.Proceedings of 2016 4th IEEE International Conference on Cloud Computing and Intelligence Systems（IEEE CCIS2016）.

[13] Yang Gao,Yan Ma, Dandan Li. Anomaly Detection of Malicious Users' Behaviors for Web Applications Based on Web Logs. Proceedings of 2017 17th IEEE International Conference on Communication Technology (ICCT 2017)