Frequent Attack Sequences-based Network log Mining

Sheng Yan\textsuperscript{1,}, Yu Chen\textsuperscript{1}, Yan Song\textsuperscript{2} and Minjie Zhu\textsuperscript{1}

\textsuperscript{1}Department of Science & Technology and ICT, State Grid Shanghai Municipal Electric Power Company, Shanghai, China
\textsuperscript{2}Electric Power Research Institute, State Grid Shanghai Municipal Electric Power Company, Shanghai, China

*Corresponding author e-mail: rcool@126.com

Abstract. As an important part of modern IT infrastructure, network provides great convenience for people to exchange information and share resources. However, network still faces with many threats such as network virus, hacker attacks, data theft and tampering and so on. Network logs includes a lot of valuable information about all behaviours happened in the network. How to analyse these network log to enhance the security of network becomes consequently the focus of many researchers. In this paper, we first design three similarity functions between two network attack records to create the network attack sequences, and then present a PrefixSpan-based frequent attack sequence mining algorithm to identify all frequent attack sequences in network log. The experimental results show that the PrefixSpan-based frequent attack sequence mining algorithm has shorter executing time and less running space than the Apriori algorithm. The PrefixSpan-based frequent attack sequence mining algorithm provides a network log analysing method for intrusion detection.

1. Introduction

With the continuous improvement of network technology, human life is becoming more and more dependent on the network. Network provides great convenience for people to exchange information and share resources. At the same time, the architecture and scale of network are getting more and more complicated with the wide application of new technologies such as Internet of Things and Cloud Computing, and the security protection of network is becoming harder and harder. How to use various novel technologies to strengthen network security becomes very important. During the running of network, massive network logs are created to record all behaviors happened in the network. These network log contain a lot of useful information for network security experts to find the network threats and strengthen the network security. Therefore, analyzing network log is very valuable to implement the active security protection of network.

For massive network log, this paper presents a network log mining algorithm based on frequent attack sequences to identify the network attack types. The main contributions of this paper are as follows:

First, a series of relativity functions are designed to calculate the correlation degree between two network attack record with a synthetic consideration on three attributes of network attack record, i.e. the occurrence time of attack, the IP address of attack and the port number of attack. These relativity functions can extract all network attack sequences in network log.
Second, a prefixSpan-based sequences mining algorithm is implemented to seek out all frequent attack sequences in network log and prove the high effectiveness of the prefixSpan-based frequent attack sequences mining algorithm in contrast to the Apriori based-frequent attack sequences by conducting multiple experiments.

The next paper structure is arranged as follows. The related works and some basic knowledge are introduced in Section 2 and Section 3 respectively. The frequent attack sequence-based network log mining algorithm is described in Section 4 and the experimental setup and result evaluation is explained in Section 5. Finally, the summary and future work are discussed in Section 6.

2. Related Works
With the rapid development of data mining technology, more and more researchers gradually realize the important role of network log analysis[1-4]. Zhang[5] proposed an attack pattern mining method using association analysis in data mining, which first divides the alert subsequences by sliding time window and then finds hidden attack patterns in network log by analyzing the alert subsequences. Hellerstein et al.[6] proposed an algorithm for mining unexpected, periodic and interdependent events from historical security logs. Treinen and Thurimella[7] designed a framework for the application of association rule mining in large intrusion detection infrastructures. Lee et al.[8] constructed an intrusion detection framework based on data mining technology and extracted the feature patterns from a large scale of audit logs, so as to discover users’ abnormal activities. Nithya et al.[9] proposed a new network log pre-processing technology that gives the concepts of local noise and global noise in data cleaning stage and cleans the network log by eliminating noise data. Sahu et al.[10] analyzed users’ past browsing behavior and proposed an improved network log mining method to predict users' browsing patterns online. Rodrigue[11] put forward the method of combining time series analysis with unsupervised learning technology to analyze the abnormal traffic log in the network, so as to provide the practical support for future network security decision-making. Bass[12] researched the fusion of intrusion detection data and multisensor data in an intrusion detection system.

The existing methods are still lacking in accuracy of mining results. Due to the lack of accuracy, there are some difficulties in the analysis of the results. At the same time, some methods only consider the time factor in mining, which is easy to cause the scene splitting of attack pattern mining results, affecting the practicability of the method.

3. Basic Concepts
In data mining, sequence data is a similar but somewhat different concept from itemset data. A comparison between itemset data and sequence data is shown in Table 1.

| TID | itemset data | SID | sequence data |
|-----|--------------|-----|---------------|
| 1   | A₁,A₂,A₃    | 1   | A₁(A₁A₂)(A₁A₂A₃)A₄ > |
| 2   | A₁,A₁,A₃    | 2   | A₂(A₂A₃)(A₂A₃A₄)(A₂A₅) |
| 3   | A₁,A₃,A₆    | 3   | A₃(A₃A₅)(A₃A₅A₇)(A₃A₇) |
| 4   | A₂,Aₐ      | 4   | A₄(A₄A₅A₇)(A₄A₇) |
| 5   | A₃,A₅,A₇   | 5   | A₅(A₅A₆)(A₅A₇) |

An itemset consists of multiple items that have no chronological order, while a sequence is constituted by several itemsets and has chronological order. For example, the first sequence <A₁(A₁A₂)(A₁A₃)(A₁A₂A₃)A₄ > is constituted by five itemset data, i.e. A₁, A₁A₂, A₁A₃, A₁A₂A₃, A₄. Subsequence is similar to the subset in mathematics. If all itemsets of a sequence A are included in the itemsets of the other sequence B, then A is a subsequence of B. The formal definition of subsequence is shown as follows:

Given that two sequences A={a₁,a₂,...,aₙ} and B={b₁, b₂,...,bₘ}(n≤m), if there exists a digit sequence t₁≤t₂≤...≤tₙ≤m that satisfies a₁⊆b₁, a₂⊆b₂,..., aₙ⊆bₙ, then A is a subsequence of B. If a₁=b₁, a₂=b₂,..., aₙ=bₙ and aₙ⊆bₙ, then A is a prefix of B.
4. Frequent Attack Sequences-Based Network log Mining

Frequent attack sequences-based network log mining includes three steps. First, the network attack sequences should be extracted from the network log. Second, the frequent attack sequences should be mined with some association rules mining algorithms. Finally, the network attack types will be clustered with some clustering algorithms. The procedure of frequent attack sequence-based network log mining is shown in Fig 1.

![Figure 1](image_url)  
**Figure 1.** The network log mining procedure based on frequent attack sequences.

4.1. Constructing Network Attack Sequences from Network log

By collecting four years' network log data from a Cisco firewall, we obtain a lot of network log. The original format of network log is show as follows:

```
<257>Oct 05 2014 22:58:37: % ASA-6-302014: Teardown TCP connection 6756 for outside:119.42.233.242/80 to inside:192.168.1.2/2211 duration 0:00:03 bytes 1367 TCP FINs.
```

These network log can be stored in a relational database with some word segmentation algorithms. Assuming \( A=\{a_1,a_2,\ldots,a_n\} \) is a network attack record extracted from the network log, and \( a_i (1 \leq i \leq n) \) is an attribute representing a kind of network attack information such as occurrence time of attack, protocol type of attack, source IP of attack, source port number of attack, destination IP of attack, destination port number of attack, duration time of attack and so forth. To seek out all network attack sequences from the network log, we design a similarity calculation function between two network attack records that takes synthetically the occurrence time of attack, the IP address of attack and the port number of attack into account.

4.1.1. The similarity calculation function of occurrence time of attack

When the time stride between two network attack records is bigger than a given time sliding window, we consider that two network attack records are not belong to the same network attack sequence. On the contrary, we consider that two network attack records are belong to the same network attack sequence and use the Gaussian distribution to compute the time similarity of two network attack record. The similarity calculation function of occurrence time of attack is as follows:

\[
S_{\text{time}}(A_i, A_j) = \begin{cases} 
  e^{-\frac{\Delta t^2}{2}}, & \Delta t \leq W \\
  0, & \text{otherwise} 
\end{cases} 
\]

\[ \Delta t = |A_i.\text{time} - A_j.\text{time}| \]  

4.1.2. The similarity calculation function of IP address of attack

We use \( L(ip_i, ip_j) \) to describe whether two IP addresses are the same, and the value of \( L(ip_i, ip_j) \) is as follows:

\[
L(ip_i, ip_j) = \begin{cases} 
  1, & ip_i \text{ is the same as } ip_j \\
  0, & \text{otherwise} 
\end{cases} 
\]

The similarity calculation function of IP address of attack is as follows:

\[
S_{ip}(A_i, A_j) = L(A_i, srcIP, A_j, srcIP) + L(A_i, desIP, A_j, desIP) + L(A_i, srcIP, A_j, desIP) + L(A_i, desIP, A_j, srcIP) / 4 
\]
4.1.3. The similarity calculation function of port number of attack

Similar to the IP address of attack, we use \( L(\text{port}_i, \text{port}_j) \) to describe whether two port number are the same, and the value of \( L(\text{port}_i, \text{port}_j) \) is as follows:

\[
L(\text{port}_i, \text{port}_j) = \begin{cases} 
1 & \text{port}_i \text{ is the same as } \text{port}_j \\
0 & \text{otherwise}
\end{cases}
\] (4)

The similarity calculation function of port number of attack is as follows:

\[
S_{\text{port}}(A_i, A_j) = L(A_i.\text{srcPort}, A_j.\text{srcPort}) + L(A_i.\text{desPort}, A_j.\text{desPort}) + L(A_i.\text{srcPort}, A_j.\text{desPort}) + L(A_i.\text{desPort}, A_j.\text{srcPort}) / 4
\] (5)

Taking the above formulas into account, we get the similarity calculation function of two network attack records is as follows:

\[
S(A_i, A_j) = \sum w_k \times S_k(A_i, A_j)
\] (6)

where \( k \) denotes respectively time, ip or port, and \( w_{\text{time}} + w_{\text{ip}} + w_{\text{port}} = 1 \).

4.2. Mining Frequent Attack Sequences from Network Attack Sequences

To identify the network attack type, we need firstly mine all frequent attack sequences from the network attack sequences. Many algorithms can be used to mine the frequent attack sequence, e.g. Apriori, Apriori-all, Apriori-some, FP-Growth, DSP and PrefixSpan. In consideration of the mining time and space, we choose PrefixSpan to find out all frequent attack sequences. The procedure of mining the frequent attack sequences with the PrefixSpan algorithm is as follows:

Input: the network attack sequence set \( S_A \), the minimum support threshold \( \text{Min}_\text{Supt} \).

Output: the frequent attack sequences \( FSA \) in \( S_A \).

Procedure:

Step 1: Scan each network attack sequence in \( S_A \) to find out all network attack sequence prefixes that the length is 1 and create the corresponding projective network attack sequence set \( 1-S_A \).

Step 2: Count the support (i.e. frequency) of each network attack sequence prefix. Delete all network attack sequences with the prefixes that the support is less than \( \text{Min}_\text{Supt} \) from \( S_A \) and obtain the frequent one-network attack sequence set \( 1-S_A \).

Step 3: For each network attack sequence prefix that the length is \( i \) and the support is bigger than \( \text{Min}_\text{Supt} \), execute the following recursively mining operation:

a) Find out the corresponding projective network attack sequence set \( i-S_A \). If \( i-S_A \) is empty, then end the recursive operation and return 0;

b) Count the support of each network attack sequence in \( i-S_A \). If the support of each network attack sequence is less than \( \text{Min}_\text{Supt} \), then end the recursive operation and return 0;

c) Join each network attack sequence with the current prefix and obtain a series of new prefixes.

d) Do \( i=i+1 \) and take each new joint prefix as a prefix to execute recursively the step 3.

4.3. Filtering All Frequent Attack Sequences with Two Rules

After mining all frequent attack sequences, we should verify these frequent attack sequences with two rules. The procedure is as follows:

Rule 1: For each frequent attack sequence, its all non-empty subsequences are still frequent attack sequence.

Rule 2: For each non-empty subsequence \( \text{sub-FS}_A \) of each frequent attack sequences \( i-FS_A (1 \leq i \leq n) \), \( \frac{\text{Support count(sub-FS}_A)}{\text{Support count(i-FS}_A)} \geq \text{Min}_\text{Conf} \) should be satisfied. Here, \( \text{Min}_\text{Conf} \) is the minimum confidence.

All frequent attack sequences can be mined from the network attack log based on the above algorithm.

5. Experimental Results and Analysis

Two metrics are often used to evaluate the mining result: \textit{precision} and \textit{recall}. Suppose \( TP \) is the number of network attack sequences that are truly frequent and mined by the above algorithm, \( FP \) is the number of network attack sequences that are truly not frequent but mined by the above algorithm, \( FN \) is the number of network attack sequences that are truly frequent but not mined by the above algorithm.
algorithm, \( TN \) is the number of network attack sequences that are truly not frequent and not mined by the above algorithm. The computational formulas of precision and recall are shown as follows:

\[
\text{precision} = \frac{TP}{TP + FP} \\
\text{recall} = \frac{TP}{TP + FN}
\]

To validate the effect of the network log mining algorithm based on frequent attack sequences, a series of experiments have been done. In the experiments, we chose about four years' network log with 1358 users and 29765 sessions from a Lenovo ThinkServer TS250 System with Intel Xeon CPU E3-1225 V5@3.3GHz, and eliminated the sessions that the frequency of occurrences is less than 0.001. Furthermore, the degree of support and confidence in two algorithms are set to 0.5 and 0.6, the threshold of time sliding windows is set to 10ms.

![Figure 2. The precision of two algorithms.](image1)

![Figure 3. The recall of two algorithms.](image2)

We adopt the precision and recall to verify the usefulness of the frequent attack sequence-based network log mining algorithm and made a comparison with the Apriori algorithm. The precision and recall of two algorithms are respectively shown in Fig. 2 and Fig. 3. On the basis of the experimental results, the precision of the PrefixSpan-based network log mining algorithm is evidently higher than the Apriori-based network log mining algorithm, and the recall of the PrefixSpan-based network log mining algorithm is slightly higher than the Apriori-based network log mining algorithm. These experimental results illustrate the favorable effect of the frequent attack sequence-based network log mining with the PrefixScan algorithm.

In addition, the PrefixSpan-based network log mining algorithm is suitable for both the sparse dataset and the dense dataset and especially the sparse dataset, while the Apriori-based network log mining algorithm is only suitable for the sparse dataset. The PrefixSpan-based network log mining algorithm holds shorter mining time and smaller mining space than the Apriori-based network log mining algorithm.

6. Conclusions and Future Work
The network log mining is important for the network security protection. Aiming at the massive and complex network log, this paper presented a PrefixScan-based network log mining algorithm to identify the frequent attack sequences and offer a network log analyzing means for intrusion detecting and security precognition. The experimental results showed the PrefixSpan-based network log mining algorithm possesses higher precision and recall than the Apriori-based network log mining algorithm. We will explore how to implement the proposed algorithm in a network security forewarning system in the next works.

References
[1] S. Tricaud, P. Saadé, Applied parallel coordinates for logs and network traffic attack analysis, Journal of Computer Virology and Hacking Techniques, 2010, 6(1):1-29.
[2] K.B. Naukudkar, D.D. Ambawade, J.W. Bakal, Enhancing performance of security log analysis using correlation-prediction technique, Proceedings of International Conference on ICT for Sustainable Development, Springer, Singapore, 2016.
[3] F. Bodon, A fast Apriori implementation, Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations (FIMI’03), Melbourne, Florida, USA, 2003, pp.3-14.

[4] L. Rokach, A survey of Clustering Algorithms, Data Mining & Knowledge Discovery Handbook, 2009, 16(3):269-298.

[5] A.F. Zhang, Z.T. Li, D. Li, et al., Discovering novel multistage attack patterns in alert streams, Proceedings of International Conference on Networking, Architecture and Storage (NAS2007), 2007, pp.115-121.

[6] J.L. Hellerstein, S. Ma, C.S. Perng, Discovering actionable patterns in event data, IBM Systems Journal, 2002, 41(3):475-493.

[7] J.J. Treinen, R. Thurimella, A framework for the application of association rule mining in large intrusion detection infrastructures, Recent Advances in Intrusion Detection, Springer Berlin Heidelberg, 2006, pp.1-18.

[8] W. Lee, S.J. Stolfo, K.W. Mok, Mining audit data to build intrusion detection models, Proceedings of International Conference on Knowledge Discovery & Data Mining, 1998, pp.66-72.

[9] P. Nithya, P. Sumathi, Novel pre-processing technique for Web log mining by removing global noise and Web robots, Proceedings of National Conference on Computing and Communication Systems, IEEE Press, 2012, pp.1-5.

[10] S. Sahu, P. Saurabh, S. Rai, An enhancement in clustering for sequential pattern mining through neural algorithm using Web logs, Proceedings of International Conference on Computational Intelligence and Communication Networks, IEEE Press, 2014, pp.758-764.

[11] A.C. Rodriguez, M.R.D.L. Mozos, Improving network security through traffic log anomaly detection using time series analysis, Proceedings of the International Conference on Computational Intelligence in Security for Information Systems, DBLP, 2010, pp.125-133.

[12] T. Bass, Intrusion detection systems and multisensor data fusion, Communications of the ACM, 2000, 43(4):99-105.