An Alert Correlation Analysis Oriented Incremental Mining Algorithm of Closed Sequential Patterns with Gap Constraints

He Hui, Wang Dong, Zhang Weizhe, Zhang Hongli

School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China, 150001
hehui@hit.edu.cn

Abstract. This paper focuses on picking up alert information efficiently and timely, which is an important need. According to the characteristics of intrusion detection log, we put forward the method of using incremental mining algorithm of closed sequential patterns with gap constraints – cispan algorithm to analyze the growing log database, we also compare the performance of cispan algorithm, prefixspan algorithm and clospan algorithm in analyzing intrusion detection log, and proves that cispan algorithm has higher efficiency in analyzing alert log.

Keywords: alert analysis, sequential pattern, gap constraint, closed sequential pattern, incremental mining algorithm of closed sequential patterns.

1 Introduction

The well-known BT site - Mininova has suffered a large-scale botnet attack across three continents recently. How to effectively and timely pick up useful alert information from the large amount of alert log data is a complicated and meaningful work.

Many complicated intrusions have a fixed time sequences, such as when a hacker attacks, at first he often scans port, executes some specific codes to get special permission, carries out an attack [1] and so on, these acts will leave the same alert sequences in alert log. Picking up alert sequences above has an important guiding significance to analyze the true purpose of intruder. In order to accurately pick up the alert sequence, incremental mining algorithm of closed sequential patterns with gap constraints – cispan algorithm will be used in this paper to the analysis of alert log, it find high frequency of frequent sequential patterns, then analyzes the contract between the alert information.

In section 2, the related definitions and content of cispan algorithm are given, in section 3, the performance comparison of cispan algorithm, clospan algorithm and prefixspan algorithm are presented. Finally, our conclusions of this paper are summarized in section 4.
2 Cispan Algorithm with Gap Constraints

Yan [2], et al. used a new pruning method on the basis of prefixspan algorithm, he found two sequences $s$ and $s'$, if $s \subseteq s'$ and $I(D_s)=I(D_s')$, then for any of item $C$ in project set $D_s$, $support(s \cup C) = support(s' \cup C)$. According to the above findings, he proposed two pruning methods – backward sub-pattern pruning method and backward super-pattern pruning method. When extending the sequence $s'$, at first we determine whether there is a sequence $s$ that has extended, which makes (1) $s' \subseteq s$ or (2) $s \subseteq s'$, if so ,we can stop extending sequence. When condition meets (1), we can directly stop extending $s'$. When condition meets (2), we don’t extend $s'$, instead, we directly transplant the offspring of $s$ to the offspring of $s'$.

When combining with gap constraints, the paper [3] put forward a prefixspan algorithm with gap constraints, when extending an element, the author puts forward a method, which records all the positions that the element appears in each of sequence in the database, rather than records the first positions that the elements appears in each of sequence in the database, but each of sequence in the sequential patterns database can only increase support degree count of the element at most once.

Cispan algorithm is future improved on the basis of clspan algorithm to speed up the speed of mining incremental database. Cispan algorithm divides the incremental operation into two steps – remove and insert. When sequence $s$ grows for $s'$, cispan algorithm first removes sequence $s$, then inserts sequence $s'$. Let $I$ be the inserted sequence, let $R$ be the removed sequence, let $U$ be the unchanged sequence. Let $IS$ be the frequent sequence that appears in $I$, let $Li$ be the prefix case that contains all sequences in $IS$. Let $US$ be the frequent sequence that appears in $U$, let $Lo$ be the sequence that contains all sequences in $US$.

Cispan algorithm [5] is divided into three steps:

1) For each frequent sequence that appears in $I$, we call incclspan algorithm to mine $Li$

2) Modifying $Lo$ of the original database. When a sequence appears in $R$, reducing the count. When the count is less than the minimum support degree, removing corresponding node of this sequence, finally we get $Lo'$

3) Merging $Lo'$ and $Li$ recursively. During merging, recursively traversing each node in preorder. Since $Li$ is the new inserted prefix case, when the structure of corresponding node of $Lo'$ is not same as that of $Li$, modifying the corresponding node of $s$. Step 2 and step 3 detailed see the cispan algorithm [10] of Ding, Yuan, et al.

3 Experimental Results and Analysis

Experimental data is the testing sample - Lincoln Laboratory Scenario(DDos) 1.0 [4] which is provided by DARPA 2000, experimental platform environment is 4 cores inter(R) Xeon(TM), CPU frequency is 3.2 GHZ, memory is 4G, OS is Linux, kernel version is 2.6.9, compiler is GCC 3.2.3, programming language is c++.
We compare the spending time of prefixspan algorithm, clospan algorithm and cispan algorithm, and separately analyze the logs that their log number are 1681, 85930 and 688134. For each log we divide the log into four paragraphs to simulate incremental process, at first use program to analyze the first paragraph log, then use the second paragraph log as increment, make program continue to analyze the new generation of log database, and use the third and the fourth paragraphs of log as increment in turn with program analyzing. Each paragraph’s increment of three logs are 100, 5000 and 20000. The results are shown in figure 1, figure 2 and figure 3, in first processing time which three algorithms need is similar, in the later log growth, cispan algorithm only needs to analyze the new incremental log, which greatly accelerates the processing speed, however, clospan algorithm and prefixspan algorithm have to deal with the whole sequential patterns database again. All in above confirm that in analyzing the intrusion detection log, cispan algorithm based on increment is better than prefixspan algorithm and cloapan algorithm in efficiency.

4 Conclusions

This paper use the cispan algorithm with gap constraints to analyze snort log, analyze the growing log database, the performance is better than the primitive clospan algorithm and prefixspan algorithm, at the same time, real-time alert performance and vast log information compression are improved to a certain extent.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (NSFC) under grant No. 61173145, the National High Technology Research and Development Program of China under Grant No. 2010AA012504 and No. 2011AA010705, and the National Basic Research Program of China under Grant No.G2011CB302605.

References

2. X. Yan, J. Han, and R. Afshar. CloSpan: Mining Closed Sequential Patterns in Large Databases. In SDM’03, San Francisco, CA, May 2003.