Design and Implementation of Intelligent Analysis Module in Database Audit System

Wei Li, Chenying Liu

North China Electric Power UniversityBeijing, China

jellybaobao@139.com

Keywords: SQL, Database Audit System, Abnormal data

Abstract. According to the current method of database auditing and the characteristics of audit data, the module detects and analyzes database operation behavior by mining technology based on association rules, establishes normal user behavior patterns, and compiles anomaly detection algorithm to detect abnormal behavior; The research, through anomaly-based detection technology, achieves lexical and syntactic analysis of SQL statement, establishes SQL statement rule base, and detects abnormal SQL statement by pattern matching of audit data. The module combines two detection direction to achieve the intelligent analysis of audit data and comprehensive detection of abnormal data.

Introduction

With the rapid development of the Internet and computer technology, database has been widely used, which is playing an increasingly important role in people's work and life. Database is the core of information system, whose security is particularly important. With the improvement of the performance and efficiency of database, how to effectively prevent database from attacks, thus ensuring the safety and validity of data in the database, has become an important research topic in information security.

In the database audit system, data auditing rules are formulated by the auditors, where some limitations and possible misuses are inevitable. Intelligent analysis module, through the analysis of audit data in learning phase, can establish and continuously update rule base, thus achieving comprehensive detection of abnormal data and averting audit omissions.

Summary of detection technology

Anomaly-based detection methods. There are two common detection technologies, one based on the detected abnormality and the other based on misuse.

The working principle of detection based on abnormality is that, through the establishment of normal behavior models, abnormal behaviors can be detected by matching the object data with behavior models. This detection method needs to undergo a period of learning phase to build the model, and then enter the data analysis phase. In the analysis phase the model still needs to be updated constantly to reduce the false detection rate. Anomaly-based detection methods are adopted in the intelligent analysis module.

Data mining technology based on association rules. Data mining and detection method based on association, through automatically searching specially-related information hidden in large amounts of data, is to extract feature model of the information, and then to determine the legality of the current user behavior based on these features.

Association rules analysis discovers valuable associations or related links in large data sets, for which the implication as $X \rightarrow Y$ is established, where X and Y are called antecedent and consequent of the association rules. Association rules are generally applied in transaction database, where every transaction consists of a record collection. Such transaction database usually includes very large amounts of data, for which current detecting techniques based on association rules are working on narrowing search space by recording certain support. Common algorithms of association rules are Apriori algorithm, partitional algorithms, FP-tree frequency set algorithm and so on.

Apriori algorithm. The Apriori algorithm is widely applied to the field of network security, such as intrusion detection system. The abnormal behavior pattern of network user can be detected by mode learning and training. The Apriori algorithm using support achieves limited and tangible mining results, enables network intrusion detection system to detect users' behavior patterns and locks the attackers on the instant, thus improving the detection efficiency based on association rules.

The Implementation of intelligent analysis

The module of user behavior audit. User behavior detection module is to detect and analyze operation behavior in the database by mining technology based on association rules, to establish a base of normal user behavior patterns and to detect abnormal behavior by compiling anomaly-detecting algorithm;User behavior detection module has two phases, learning phase and testing phase.

Learning Phase: The user sets the learning time. The longer the learning time, the more improved the rule base. In the learning phase, user behavior detection module extracts and records the collected audit data. After the learning phase, it analyzes records of the data by the Apriori algorithm, discovers the frequent rules, and then defines strong association rules in all the frequent rules. Flow chart is as follows:

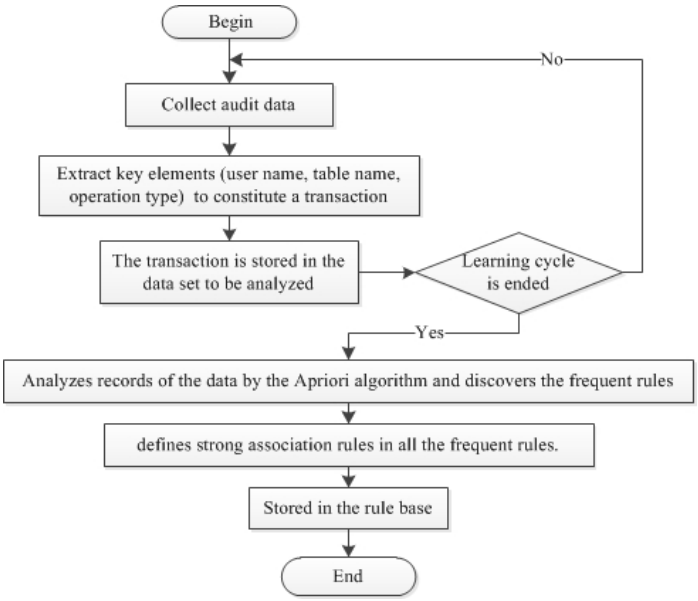

Fig. 1. Process of learning phase in user behavior audit

Here is a simple example to demonstrate how Apriori algorithm discovers frequent item sets of data to be analyzed and generates strong association rules:

Assume that the data to be analyzed are as follows:

Table 1. Transaction Table

Transaction	Username	Target Table	Operation
D1	User1	Table1	Insert
D2	User1	Table2	select
D3	User1	Table2	delete
D4	User2	Table1	select
D5	User2	Table1	select
D6	User2	Table1	insert
D7	User3	Table1	select
D8	User3	Table2	insert
D9	User3	Table2	select
D10	User3	Table2	select

The procedure of using Apriori algorithm to find all frequent item sets is as Figure 2:

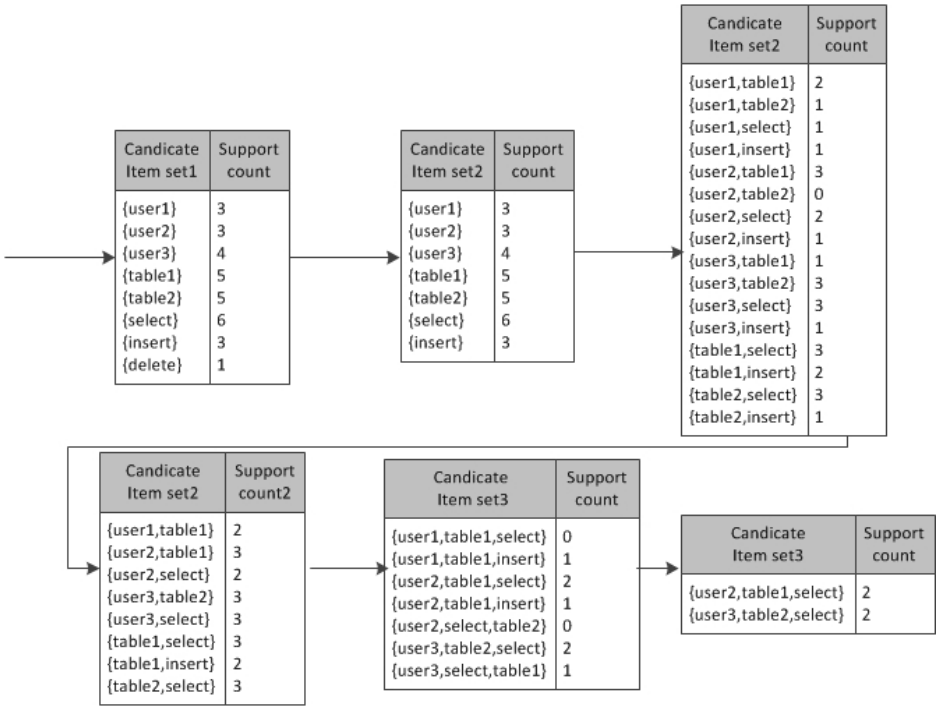

Fig. 2. Process of apriori algorithm

The methods of generating strong association rules from frequent item sets are as follows:

Frequent item sets are {user2, table1, select}. Its non-empty subsets are {user2, table1}, {user2, select}, {table1, select}, {user2}, {table1} and {select}. Corresponding confidence is as follows:

user2 && table1 -> select confidence=2/3=66%
user2 && select ->table1 confidence=2/2=100%
table1 && select -> user2 confidence=2/3=66%
user2 -> table1 && select confidence=2/3=66%
table1-> user2 && select confidence=2/5=40%
select -> user2 && table1 confidence=2/6=33%

If min_conf=60%, then strong rules are user2&&table1->select, user2&&select>table1, table1&&select->user2, user2 ->table1&&select.

Similarly, the strong rules of frequent item sets {user3，table2，select} are user3&&table2->select，user3&&select>table2, table2&&select->user3.

Testing Phase: In the detection phase, extract the key elements of audit data as transactions (user name, the operation object table name, operation type), and match them with the normal user behavior base:

a. If the match successes, define the data as normal behavior and the detect ends;

b. If the match fails, then alarm. And the data are recorded. Set counting statistics as the same transactions are recorded. When the count exceeds the set threshold, analyze these transactions together with the historically learned data to calculate the confidence of that rule. If the confidence degrees meet the set threshold, it is submitted to the auditors. The database is updated if it passes the audit. If not, the data is returned.

The module of SQLstatement audit. Learning Phase: In the learning phase, SQL statement detection module abstracts SQL statement from collected audit data, establishes SQL syntax tree, computes its feature V(S1,S2, S3... Sn), and finally stores the feature V and SQL syntax tree as a rule into the SQL statement rule base.The process is as follows:

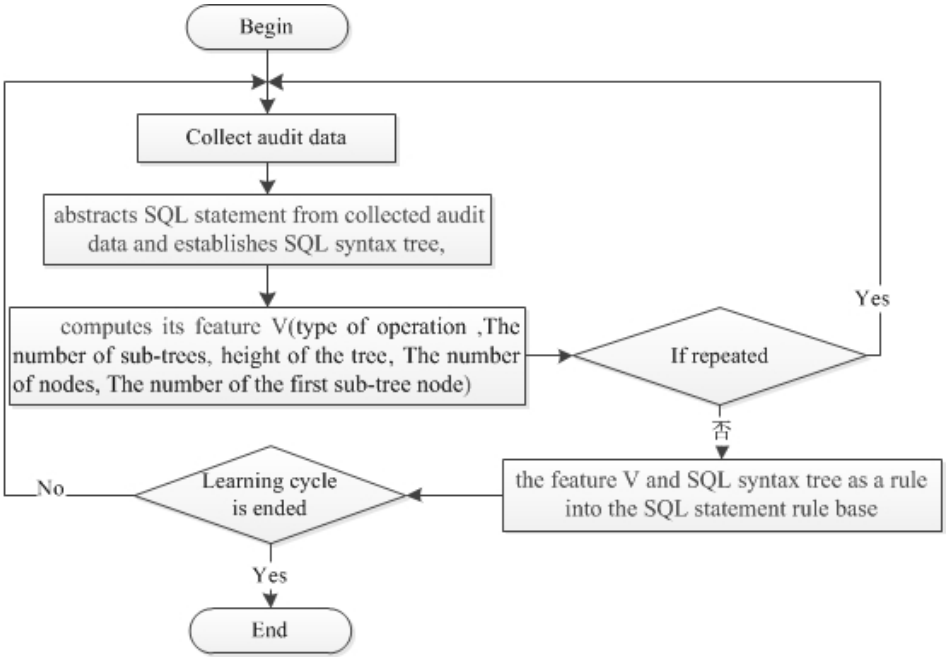

Fig. 3. Process of learning phase in SQL statement audit

Here is a simple example to demonstrate how the SQL statement generates syntax tree and calculates the feature:

Assuming that the SQL statement is: select a from b where username=1and password=2

SQL statement detection module processes SQL statement by lexical analysis tools LEX and syntax analysis tool YACC provided by UNIX system to realize the lexical and syntactic analysis, generates syntax tree including all SQL information. After the replacement of the user input with a unified symbol (In this case, # is the replacement for 1, and $ is the replacement for 2.), specification syntax tree is as follows:

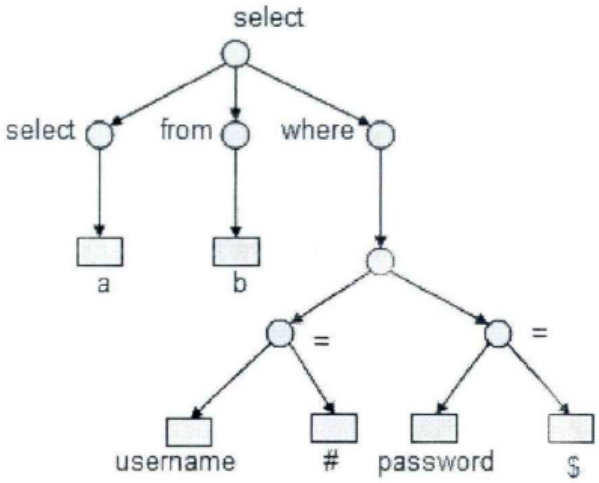

Fig. 4. SQL syntax tree

The extracted features from the structure SQL specification syntax tree include:

S1: type of operation

S2: The number of sub-trees

S3: height of the tree

S4: The number of nodes

S5: The number of the first sub-tree node

They are expressed in the above SQL specification syntax tree as V (i) = V (2, 3, 4, 13,2).

Store V (i) in the rule base.

Testing Phase: In the anomaly detection, through pattern matching, SQL statement detection module looks up entries in rule database by V(i) as index, finds the SQL statement with the same feature as V(i), and then matches every node of syntax tree one by one to define whether or not it matches the SQL statement. If the matching successes, the SQL statement is normal and the process ends; if not, the SQL statement will be submitted to the manual audit.

Conclusions

Through the analysis of user behavior and SQL statement, the research designs an intelligent analysis module in database audit system. Mixing the anomaly detection technology and the mining technology based on association rules, it achieves automatic learning and detecting of database audit rules. The module in database audit system has good universality and high efficiency in the detection, but the module requires learning phase with high learning cost because of the numerous database users. Even after a period of learning, into the detection phase, auditors are still indispensable in determining the abnormal rules and update the rule base frequently. Not timely-updated rules may lead to an increased rate of false detection.

References

[1] T.H. Liu, H.F. Zhu, Chang Guiran, et a1． "The design and implementation of zero－copy for linux" Eighth International Conference on Intelligent Systems Design and Applications. 2008: 121－123.

[2] Welsh M., Basu A., von Eicken T.. "Incorporating memory management into user level network interfaces." Cornell University Ithaca , NY, USA: Technical Report TR9721620 , 1997

[3] S.Q. Wang, D.S. Xu, S.L. Yan. 2010. “Analysis and application of Wireshark in TCP/IP protocol teaching”. E-Health Networking, Digital Ecosystems and Technologies (EDT), 2010 International Conference on (Volume: 2).

[4] Fusion embedded TCP/IP stack, http://www.unicoi.com/fusion net/fusion tcpip.htm.

[5] Navaro G R M. "Flexible Pattern Matching in Strings". Cambridge: Cambridge University Press,2002

[6] C.S. Miao, G.R. Chang, X.W. Wang． "Filtering Based Multiple String Matching Algorithm Combining q-Grams and BNDM" Proceedings of the 2010 Fourth International Conference on Genetic and Evolutionary Computing. ACM, 2010: 582—585