Framework to Detect Malicious Transactions in Database System

Dhanashree Parchand
ME Student of Department of Computer Engineering MMCOE, Pune

H. K. Khanuja
Department of Computer Engineering MMCOE, Pune

ABSTRACT

Every organization is associated with huge amount of information which is more valuable. Data is important and so it should be consistent, accurate and correct. Today many approaches are used to protect the data as well as networks from attackers (attacks like SQLIA, Brute-force attack). One way to make data more secure is using Intrusion Detection System (IDS). Many researches are done in this intrusion detection field but it mainly concentrated on networks and operating system. This approach is for database so that it will prevent the data loss, maintain consistency and accuracy. Database security research is concerned about the protection of database from unauthorized access. The unauthorized access may be in the form of execution of malicious transaction and this may lead to break the integrity of the system. Banking is one of the sectors which are suffering from million dollars losses only because of this unauthorized activities and malicious transactions. So, it is today's demand to detect malicious transactions and also to provide some recommendation. In this paper, we provided the detection system for the real-world problem of intrusion detection in the banking system and we are going to give some preventive measures to avoid or reduce future attacks. In order to detect malicious transactions, we used data mining algorithm for framing a data dependency miner for our banking database IDS. Our approach extracts the read-write dependency rules and then these rules are used to check whether the coming transaction is malicious or not. Our system not only finds the malicious transactions that corrupt data items but also identify the transactions that write data without permission and read data without permission.

Keywords

Intrusion Detection, Malicious Transaction, Data Mining, Data Dependency.

1. INTRODUCTION

In this era it is very important to secure database system from any malicious activity. Information is one of the main assets of any organization which is essential to its continuity. Therefore, information security is very important to protect the confidentiality, integrity and availability of the information. Many systems and tools are used to achieve the needs of the data security and to prevent database systems from any possible incident. Some data stored in databases is very expensive and valuable, but existing security models are insufficient to prevent misuse of that data, especially insider abuse by legitimate users [3]. Hence, a growing number of researches have concentrated on handling the vast variety of malicious attacks to theses data. Malicious attacks to database systems may cause corrupted data and this data can spread very fast to other parts of the database system through the authenticated users. Especially some irresponsible people are always there to challenge the robustness of a banking system. So banking organizations need to be able to detect the intrusions into their network and database systems as soon as possible so that they can prevent further damage to sensitive data.

Our proposed system first extracts only interesting read and write dependency rules from all the log files of normal transactions and these extracted rules are used to verify whether incoming transaction is malicious or not. This system not only extracts dependency between data items, but also considers the access operations of these data items (read or write) in extracting rules. The extracted rules are used to verify the malicious activity.

1.1 Motivation

From real-world database application; it is observed that although the transaction program changes often, the whole database structure and essential data correlation rarely changes.

So, the idea of using this dependency among data items for detecting intrusion transaction in database came into picture. In now days, database system is very important for most of the organization where information play an important role. Database size has grown in different ways: the number of records, or objects in the database and the number of fields or attributes per object [4]. As a result, it is difficult for administrators to keep track of whether the attributes are being accessed only by genuine transactions or it is done by some unauthorized users. Specially in banking database, all data is valuable and should be protected from any of the attack like SQL injection attack, brute force attack. SQLIA is technique where the attacker changes the structure of the query by injecting input in the query; which is formed by the programmer and gaining the access of database which results modification or deletion of user's data. Existing data mining based IDSs use support-confidence framework for extracting dependencies among data items. The major limitation of these existing approaches is that the accuracy in setting up minimum support directly influences the number and the quality of the discovered classification rules [1]. So to overcome these limitations, our system which extracts only the interesting rules that have lower frequency than minimum support threshold is used and this system manages the number of extracted rules and avoids extracting uninteresting rules as we are interested in only few rules which are based on both access type and order of the data items.

2. RELATED WORK

Intrusion is defined as any set of actions that attempt to compromise the integrity, confidentiality or availability of a resource [1]. IDSs involve determining the malicious actions performed against computer systems and alerting the system administrator. The existing IDSs can be divided into two categories: signature-based and anomaly based systems. In signature-based systems, a set of signatures are manually constructed by security experts analyzing previous attacks and the collected signatures are matched against currently occurring activities to detect intrusions [1]. Clearly signature-based IDS cannot detect unknown attacks without any pre-collected signatures. In 2004, Hu and Panda proposed a data mining approach and they implemented a model for detecting malicious transactions that are targeted at corrupting data [3]. They have used a data mining approach for mining data dependencies among data items, which are in the form of classification rules [3]. In 2008, Hashemi et al. proposed approach is based on (1) mining dependencies among data items, (2) finding abnormal update patterns in time series corresponding to each data item's update history, and (3) using a behavior similarity criterion between normal transactions and the transactions in question [4]. Their approach has three advantages. First, dependency rules among data items are extended not only for detecting transactions that write data without permission, but also for detecting transactions that read data without permission. Second, a novel behavior similarity criterion is introduced to reduce the false positive rate of the detection. Third, time-series anomaly analysis is conducted to identify intrusion transactions, which update data items with unexpected pattern [1]. In 2010, Hu and Panda proposed inter-transaction data dependencies approach; in which they have considered that the malicious transactions launched by the attacker can be so well crafted that an attacker may launch a group of malicious transactions each of which appears as a normal transaction [2]. They have found that it is critical to develop an algorithm that clusters user transactions into user tasks for facilitating discovery of inter-transaction data dependencies. They have defined a user task as a group of transactions that are semantically related to perform a user task and the data access operations that are not compliant with these intrinsic inter-transaction data dependencies are identified as anomalous activities [2].

3. PROBLEM STATEMENT

We propose a system which will be able to track normal transaction and detect malicious ones more effectively. It detects malicious transactions that corrupt the data. There are many types of insider attack that try to abuse the access rights and do malicious activities for example, employees masquerading and the malicious activities such as updated and deleted approved records. A malicious activity is defined as a group of actions that attempts to harm the integrity, confidentiality of database system. One of the database security problems is inside malicious activities. Among them are: updating of approved records with malicious data, and deleting approved records. This study hypothesizes that dependency relationship among data items can be used to detect and prevent the above mentioned malicious activities. It will first extract dependency between the data items and then it will consider the access operation of these data items. The aim is to develop intrusion detection mechanism using dependency relationship among data items to detect and prevent the malicious activities in database management systems. It will extract the interesting read-write dependency rules [1] and according to the rules it can decide whether the transaction is malicious or not. And it will provide some preventive measures to reduce or avoid future malicious activities.

So the specific objectives are-

1. To detect intrusion transaction in database system.
2. To prevent data items.
3. To find transaction that read without permission.
4. To find transaction that write without permission.
5. To avoid future malicious transaction.

4. PROPOSED SYSTEM

Database IDS profiles patterns of normal user transactions and uses these patterns for identifying intrusive behavior, such that the transactions not compliant with the patterns are identified as malicious. Database IDS obtains patterns of normal transactions by mining data dependencies among data items in database systems. Data dependencies discovered by the data dependency miner component are employed as rules for identifying anomalies. Data dependency miner component has an important role in data dependency based IDSs [1]. In fact the accuracy of a data dependency based IDS is based on its data dependency miner component because the rules, which are extracted by this component, are the main criterion for detecting new intrusive transactions. The input to the system is a log file of normal transactions and its output is extracted read and writes dependency rules from the log file. These extracted dependency rules represent essential correlations between data items of normal transactions and are a criterion for detecting future intrusion transactions.

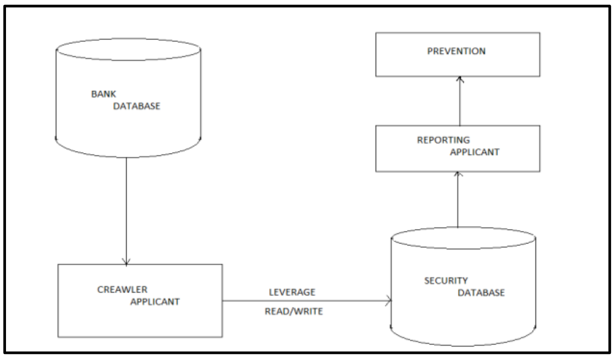

Fig1: System Architecture

In the above block diagram, there are mainly five blocks which are nothing but the modules. This diagram shows the working of our project. As this system detects intrusion transaction in the database so there must be some huge amount of database i.e. for our project banking database.

1. Bank Database:

Banking database contains all the information of the transactions i.e. date, time, Mac-id, transaction-id, account-no etc... And contains all data about employees, customer, accounts, account-types etc... This contains huge amount of data about bank which is valuable and secret. As transaction is performed log files will be generated which shows all the history. It maintains the data according to read, write transaction.

2. Crawler Application:

1. It reads each transaction from the database.
2. It categorizes the transaction in the form of Read or Write.
3. It generates Read sequence.
4. It also generates Write sequence.
5. It calculate key measure value i.e. Leverage value.
6. According to leverage value all transaction which shows abnormal behavior is transferred to security database.

To create the normal transactions we generate read and write operations based on a dependency factor. The dependency factor defines the average number of read operations immediately before a particular write operation w(x) or the average number of write operations immediately after that write operation.

3. Security Database:
According to data dependency and leverage value some transaction which shows malicious behaviors are stored into this database. The transactions in this database are not showing normal behavior so we have to analyze it.

4. Reporting Application:
As we know the malicious transaction this module generates the report on the basis of date and category. According to the date and time one record is generated which also gives Mac-id of that attacker. Database admin take one look on it and analyze it to detect the attacker and type of attack. This also generates report on the basis of category i.e. SQLIA, Brute-Force attack etc…

5. Prevention:
After analyzing that report by admin it can prevent this type of attack by giving some preventive measures like we can block that attacker as we know the Mac-id of the system to prevent any future attacks.

Fig. 2 shows the flow of our project. The step by step process of project is as:

1. Employee of the bank enters data in bank database then logs are getting generated and stored.
2. Generated database logs are export to crawler application module.
3. Then crawler reads all the transaction one by one.
4. It categorizes it into Read or Write form and generate sequence for each read-write transaction.
5. It calculates the leverage value according to the formula. Leverage measures the difference of the probability of X and Y appearing together in the data set and what would be expected if X and Y were statistically independent [1].

   Leverage(X → Y) = P(X and Y) − (P(X) × P(Y)) [1].
6. It stores all invalid transaction which shows abnormal behavior based on leverage value in security database.
7. Database administrator at the end of the day checks the report and carefully analyzes it.

Then Database administrator gives some preventive measures like block that user using Mac-id to prevent future attacks.

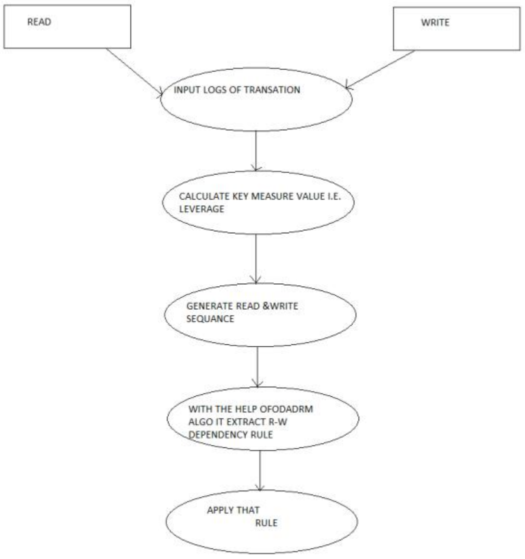
READ
WRITE
INPUT LOGS OF TRANSATION
CALCULATE KEY MEASURE VALUE I.E. LEVERAGE
GENERATE READ &WRITE SEQUANCE
WITH THE HELP OFODADRM ALGO IT EXTRACT R-W DEPENDENCY RULE
APPLY THAT RULE

Fig 2: Data Flow Diagram of Proposed Framework

5. EXPECTED RESULTS

Our work is totally focused on development of database intrusion detection system. According to [1], M. Sohrabi et al. compared two different approaches i.e. Hu and Panda method and ODADRM algorithm. They found ODADRM is better approach with respect to complexity, time, accuracy.

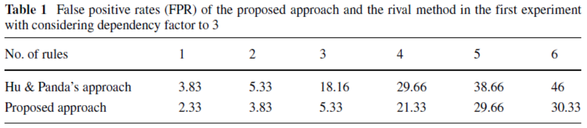
Table 1 False positive rates (FPR) of the proposed approach and the rival method in the first experiment with considering dependency factor to 3

No. of rules	1	2	3	4	5	6
Hu & Panda's approach	3.83	5.33	18.16	29.66	38.66	46
Proposed approach	2.33	3.83	5.33	21.33	29.66	30.33

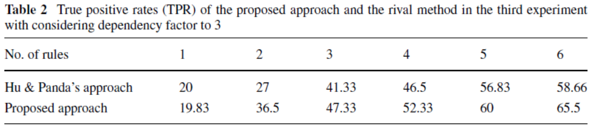
Table 2 True positive rates (TPR) of the proposed approach and the rival method in the third experiment with considering dependency factor to 3

No. of rules	1	2	3	4	5	6
Hu & Panda's approach	20	27	41.33	46.5	56.83	58.66
Proposed approach	19.83	36.5	47.33	52.33	60	65.5

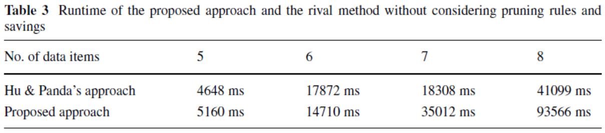
Table 3 Runtime of the proposed approach and the rival method without considering pruning rules and savings

No. of data items	5	6	7	8
Hu & Panda's approach	4648 ms	17872 ms	18308 ms	41099 ms
Proposed approach	5160 ms	14710 ms	35012 ms	93566 ms

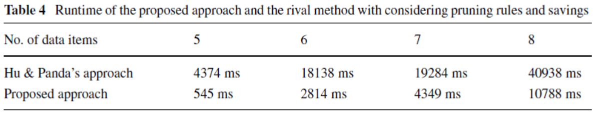
Table 4 Runtime of the proposed approach and the rival method with considering pruning rules and savings

No. of data items	5	6	7	8
Hu & Panda's approach	4374 ms	18138 ms	19284 ms	40938 ms
Proposed approach	545 ms	2814 ms	4349 ms	10788 ms

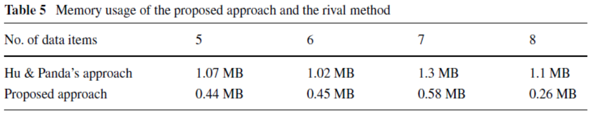
Table 5 Memory usage of the proposed approach and the rival method

No. of data items	5	6	7	8
Hu & Panda's approach	1.07 MB	1.02 MB	1.3 MB	1.1 MB
Proposed approach	0.44 MB	0.45 MB	0.58 MB	0.26 MB

From this entire above table it is clear that proposed approach is much better than previous approaches.

6. CONCLUSION

Malicious detection mechanisms play a crucial role in the security landscape of an organization. In this we have focused malicious detection system for database. Existing data mining based IDSs use support-confidence framework for extracting dependencies among data items [2]. The major limitation of these approaches is that the accuracy in setting up minimum support directly influences the number and the quality of the discovered rules. Our approach can extract interesting rules that have lower frequency than minimum support threshold. Also it manages the number of extracted rules and avoids extracting uninteresting rules. On the basis of these leverage value and dependency rules we can find the malicious transaction. There are many interesting issues to investigate as future work. For instance ODADRM only finds malicious transactions that corrupt data items and cannot identify transactions that read data without permission. This results in a significant reduction in detection rate when most malicious transactions are only reading data items illegally. We also create report which shows all malicious transaction carried out and on the basis of that we give some preventive measures to avoid future attacks.

7. REFERENCES

[1] Mina Sohrabi, M.M.Javidi, S.Hashemi “Detecting intrusion transactions in database systems: a novel approach” J Intell Inf Syst 42:619-644 DOI 10.1007 Springer 2014.

[2] Hu, Y., and Panda, B. “Mining inter-transaction data dependencies for database intrusion detection”. In Proceedings of innovations and advances in computer sciences and engineering. Springer 2010.

[3] Hu, Y., and Panda, “A data mining approach for database intrusion detection.” In Proceedings of the ACM symposium on applied computing (pp. 711716) ACM 2004.

[4] Hashemi, S., Yang, Y., Zabihzadeh, D., Kangavari, M. “Detecting intrusion transactions is databases using data item dependencies and anomaly analysis”. Expert Systems, 25(5), 460473 2008.

[5] Lee, V.C., Stankovic, J., Son, S.H. “Intrusion detection in real-time database systems via time signatures.” In Proceedings of the 6th IEEE real time technology and applications symposium (RTAS00) (pp.124133). New York: IEEE Press 2000.

[6] Srivastava, A., Sural, S., Majumdar, A.K. “Database intrusion detection using weighted sequence mining.” Journal of Computers, 1(4), 817 2006.

[7] Bertino, E., Kamra, A., Terzi, E., Vakali, A. “Intrusion detection in RBAC- administered databases.” In Proceedings of 21st annual computer security applications conference (pp. 170182) 2005.

[8] Barbara, D, Goel, R., Jajodia, S. “Mining malicious data corruption with Hidden Markov Models.” in Proceedings of the 16th annual IFIP WG 11.3 working conference on data and application security.Cambridge 2002.

[9] Chung, C.Y., Gertz, M., Levitt, K. “Demids: a misuse detection system for database systems.” In Integrity and internal control information systems: strategic views on the need for control (pp. 159178) Norwell: Kluwer 2000

[10] Webb, G.I., and Zhang, S. “K-Optimal rule discovery. Data Mining and Knowledge Discovery" 10(1), 3979, 2005.