ISSN 22773061

Forensic Analysis of Databases by Combining Multiple Evidences

Harmeet Kaur Khanuja[1], Dr. D.S. Adane[2]
[1]Department of Computer Engineering, Pune University, MS, India
harmeet.khanuja27@gmail.com
[2]Department of IT, RKNEC, Nagpur
dattaadane@yahoo.com

ABSTRACT

The information security for securing enterprise databases from internal and external attacks and violations of mutual policy is an interminable struggle. With the growing number of attacks and frauds, the organizations are finding it difficult to meet various regulatory compliance requirements such as SOX, HIPAA, and state privacy laws. The aim here is to develop a methodology which monitors the database transactions on continuous basis and to make a decision whether the database transactions are legitimate or suspicious by combining multiple evidences gathered. The suspicious transactions can then be used for forensic analysis to reconstruct the illegal activity carried out in an organization. This can be achieved by incorporating information accountability in Database Management System. Information accountability means, the information usage should be transparent so that it is possible to determine whether a use is appropriate under a given set of rules. We focus on effective information accountability of data stored in high-performance databases through database forensics which collects and analyses database transactions collected through various sources and artifacts like data cache, log files, error logs etc. having volatile or non-volatile characteristics within high performance databases. The information and multiple evidences collected are then analyzed using an Extended Dempster-Shafer theory(EDST). It combines multiple such evidences and an initial belief is computed for suspected transactions which can be further used for reconstructing the activity in database forensics process.

Indexing terms/Keywords

Database Forensics, Dempster-Shafer theory, Artifacts, Transactions, Initial Belief

Council for Innovative Research
Peer Review Research Publishing System
Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY
Vol 7, No 3
editor@cirworld.com
www.cirworld.com, member.cirworld.com

ISSN 22773061

INTRODUCTION

Information is one of the most important assets for any organization. It is essential to protect such an asset for establishing and maintaining a truthful relationship between any organization and its clients or user community. Even though considerable efforts are taken to protect organizations databases, the number of records breached by various means has grown each year [1]. According to a computer crime and security survey conducted by the Computer Security Institute (CSI) [2] in 2011, large-scale breaches dropped dramatically while small attacks increased. The X-Factor US database hack is one of the string of attacks on corporate servers to extract personal data which suggests that cybercriminals are now building information profiles on people, rather than developing frauds around available credentials. One of the most effective online fraud deterrents available are services that analyze transaction histories to establish a customer's basic use profile. When a new transaction is sufficiently outside the norm, the transaction can be subjected to further scrutiny before it's approved. A perperator counteract the effectiveness of this defense by building up their own profiles to go along with stolen credentials and making sure that fraudulent transactions were sufficiently within the norm that red flags weren't raised [3].

Traditional database security mechanisms do not assure protection against database vulnerability exploits and are limited in defending security attacks from inside the organization or from unsecured applications. Although DBMS offer basic database security features such as authentication, authorization, and access control, these alone are not satisfactory to defend against growing security threats [1]. In addition,legislation and regulatory requirements such as Sarbanes Oxley (SOX) [2,4] for public companies, HIPAA [2,5,6] in healthcare mandates that companies and organizations take certain measures to ensure the privacy, integrity and security of sensitive data. Auditing often has a considerable impact on system performance, don't offer granular compliance reports or provide end-to-end security visibility, and do not scale well in heterogeneous DBMS environments, leading to a significant requirement in people resources and cost. In addition, because basic auditing typically relies on DBMS capabilities, it is often controlled by DBAs and other privileged users. Objective of an 'audit' is to determine whether all the transactions are properly recorded in the accounts, and appropriately reflected in the organization's statement and reports. The objective of a database forensics investigation is to identify "digital evidences" using scientifically derived and proven methods that can be used to facilitate to reconstruct events in an investigation. It also aims to identify the responsible person and seriousness of the misconduct [8]. There is always a probability that privileged users can easily disable native auditing or modify audit logs to achieve the mischevious target [7]. The organizations would like to be assured that such tacts shouldn't take place and if it has occurred it should be quickly discovered and used to identify the perpetrator.

Continuous monitor of database transactions can keep accountability of the databases activities from unauthorized accesses or malicious actions carried out by an intruder. This can be assured by keeping information accountability [9,10] that is if the system administrator is able to track users' activities which is carried out in a trusted server then users can be held responsible for their illegal actions. There must be some reliable way to monitor who is performing what operations on the data. Databases preserves a historical record of activities and data which can be a rich source to offer the benefit of system accountability. A change towards information accountability presents valuable advantages over information restriction and access controls in the particular area of correct storage, use, and maintenance of databases. An information accountability approach to database security is cheaper, can protect against a variety of threats (including insider threats), can successfully deal with the consequences of information restriction failure and can provide complex security problems tractable. We are working to show information accountability can effectively realize appropriate use (i.e., guarantee no unauthorized modifications—insertions, deletions, updates) in high-performance databases[9,10]. These systems aim to detect attacks as early as possible so that the damage caused by the attackers is minimized with true positive. True positives (TP) are the intrusive transactions caught by the system and False positives (FP) are the genuine transactions marked as intrusive or called false alarms. The possibility of enhancing existing security mechanisms by introducing an effective information accountability through database forensic constitutes the objective of our work.

In this paper, we propose database forensic methodology [11] that collects and analyses evidences and artifacts (volatile and non-volatile) like data cache, log files etc. in high performance database. Since multiple evidence and artifacts are gathered we use an Extended Dempster-Shafer Theory [15] by combining theses evidences for decision making. The Dempster-Shafer theory provides a new method to analyze data from multiple sources[12, 13, 14] which can be applied to predict the probability of tampering within database. The theory's rule of combination gives a numerical method to fuse multiple pieces of information to derive a conclusion.

RELATED WORK

Database Forensic is an essential area which must need research awareness. The lack of research is due to the inherent complexity of databases that are not fully understood in a forensic context yet. It is said that databases are inherently multidimensional from a forensic perspective [11]. The paper InnoDB Database Forensics shows how the MySQL tables in the .frm files are built and how important information is saved. The aim was to identify and name the bytes and interpret them. With that knowledge, it is possible to detect inconsistencies in the database. But there was no knowledge discovered for the multiple log files and cache for further analysis [16]. A survey study is carried out in our previous work on "Database Security Threats and Challenges in Database Forensic" which highlights the work done by various researchers in the area of database forensics [17]. A tutorial work is carried out for MySQL to create framework for database forensic analysis in our paper [18]. Research work is carried out for database intrusion detection in the paper [19] which combines evidences from current as well as past behavior of users. The evidences are collected on the

transaction's behavior. An Extension of Dempster- Shafer's Theory (EDST) is used to combine multiple such evidences and an initial belief is computed. The belief is updated according to its similarity with malicious or genuine transaction history using Bayesian learning. In our work, Extended Dempster-Shafer's Theory is applied to the database transactions. The evidences through transactions are collected from database artifacts like SQL Server Error Logs and Data Cache which are further combined by EDST. The suspicious transactions discovered will be then processed along with other evidences for reconstructing the suspicious activity. This will save the time and cost in investigating the illegal activity carried out in an organization in an optimal way.

EXTENDED DEMPSTER SHAFER THEORY

Dempster-Shafer theory is a mathematical theory of evidence [12,13,14], which provides a method for combining evidences from different sources without prior knowledge of their distributions. In this method, it is possible to assign probability values to sets possibilities rather than to single event only, and it is not needed to divide all the probability values among the events, once the remaining probability should be assigned to the environment and not to the remaining events, thus modeling more naturally certain classes of problems. It is concerned with bounds for probabilities of provability rather than computing probabilities of truth. The two bounds used in Dempster-Shafer are based on belief functions and plausible reasoning. A frame of discernment (also called a Universe of Discourse) in Dempster-Shafer is a set of mutually exclusive and exhaustive possibilities denoted by 'UD'. Any hypothesis 'A' will refer to a subset of 'UD' for which observers can present evidence. The set of all possible subsets of 'UD', including itself and the null set 'Ø', is called a power set.

Two basic probability assignments (BPAs) $m_1(h)$ and $m_2(h)$ are combined by the Dempster's rule as follows:

$$m(h) = m_1(h) \oplus m_2(h) = X \sum_{x \cap y = h} m_1(x) m_2(y) \tag{1}$$

where, X is the normalization constant defined by the following Eq. (2):

$$X = \frac{1}{K} \tag{2}$$

$$K = 1 - \sum_{x \cap y = \phi} m_1(x) m_2(y) \tag{3}$$

The Dempster's rule for combination as shown in Fig. 1 is a procedure for combining the independent pieces of evidence to reach to Decision.

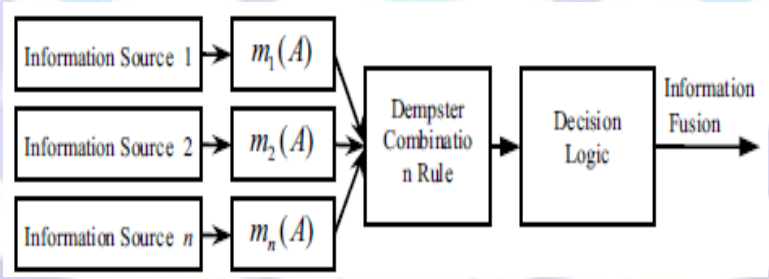

Fig 1: Dempster Combination Rule

However, the basic DST has some major drawbacks. The normalization constant in the Dempster's combination rule (Eq. (1)) has the effect of completely ignoring conflict and consequently, this operation will yield counterintuitive results in the face of significant conflict in certain contexts. To solve this problem, we have employed the Extended Dempster-Shafer theory proposed by Campos et al. [15] that presents a new improved rule for combining evidences. EDST overcomes the above mentioned pitfalls by assigning the beliefs according to the degree of conflict between the evidences and the remaining belief to the environment rather than the common hypothesis. The conflict between two belief functions bel_1 and bel_2, denoted by $Con(bel_1, bel_2)$ is given by the logarithm of the normalization constant as follows:

$$Con(bel_1, bel_2) = \log(X) \tag{4}$$

The modified Dempster's combination rule automatically incorporates the uncertainty coming form the conflicting evidences which is given by the following Eq. (5):

$$m(h) = m_1(h) \oplus m_2(h) = \frac{X \sum_{x \cap y = h} m_1(x) m_2(y)}{1 + \log\left(\frac{1}{K}\right)} \tag{5}$$

ISSN 22773061

ARTIFACT COLLECTION

To prove in the concept, the database SQL Server is studied and analyzed for this proposed methodology. We first identify the key artifacts in the high performance database from where we can find the traces of the activities carried out in an organization. Table below shows the key artifacts in SQL server. SQL Server artifacts can generally be classified as one of the two types [22]:

• Resident artifacts: Reside within files and memory locations explicitly reserved for SQL Server use, such as the SQL Server error log.

• Nonresident artifacts: Reside within files not explicitly reserved for SQL Server use. An example would be SQL Server data written within the Windows system event log.

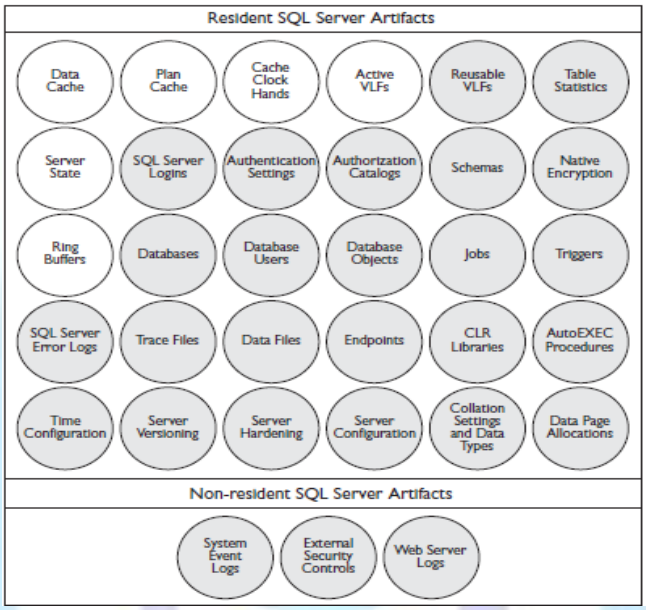

Fig 2: SQL Server Artifacts

A multitude of SQL Server artifacts are available that can provide valuable clues during a database investigation. These artifacts may hold clues that will help you piece together the incident events. So we can get multiple evidences from various artifacts.

Dempster-Shafer Rule for Combination Applied for Database Forensics

Dempster's rule for combination is a procedure for combining independent pieces of evidence. In our system for Forensic accounting the accountability for malicious activity has to be marked by identifying sign of intrusion. The collected artifacts or information sources can hold malicious transactions to uncover any active unauthorized database access. To formulate it we represent the system as as a 5-tuple { U,T,Ψ, θ_{LT}, θ_{uT},} where:

1) U = {U_1, U_2,……. U_n} is the set of users in the organization
2) T = {T(U_1), T(U_2),…… T(U_n)} is the set of transactions where each T(U_k) corresponds to the transaction of the user U_k.

Each transaction T(U_k) corresponds to the user U_k. Each transaction T(U_k) with respective to U_k can be represented as a 5 tuple: {user_ID, attribute_ID_seq, client_net_address, last_execution_time, table_ID_seq}

- User_ID: a number that identifies each user uniquely
- attribute_ID_seq: attribute access sequence in a transaction
- client_net_address: identifies the location where a transaction was carried out

  (Note: In our work client_net_address is 'local machine' since the work is carried out on single machine in future it would be extended in network)

- last_execution_time: time slot in which a transaction occurs
- table_ID_seq: table access sequence in a transaction

3) $\Psi(T_{j,\rho}^{U_k})$:suspicion score of the j[th] transaction $(T_{j,\rho}^{U_k})$ by user U_k with time gap ρ

4) θ_{LT} : lower threshold {0≤ θ_{LT} ≤1}

5) θ_{uT} : upper threshold, where {(0< θ_{uT} ≤1) ∧ (θ_{LT} ≤ θ_{uT})}

In our system a number of rules are used to analyze the deviation of each incoming transaction from the normal transactions of the user by assigning basic probabilities to it. The basic probability values are combined to obtain an initial belief by applying extended Dempster-Shafer theory[18]. In order to meet this functionality, the system is designed with two major Units, namely, Rule-based Unit (RBU) and Belief Combination Unit (BCU).

Rule-based unit: The RBU consists of a number of rules which classify transactions collected from artifacts as suspicious with a certain probability. It measures the extent to which a transaction's behavior deviates from the user's normal transaction for each new transaction. Depending upon the sign of intrusion identified we briefly discuss two of the rule-based techniques here.

Sequence Alignment for Deviation Detection (R1):

This rule is designed for database reconnaissance activity. Authorized database users are normally familiar with the layout of the database with which they are working and will often execute queries requesting specific data columns using previously developed views, procedures, and ad hoc SQL statements. In contrast, unauthorized database users are typically not familiar with the database layout and will execute broader-scope commands intended to reveal the structure of the database and the type of data stored within it. This activity is referred to as database reconnaissance. To show the intrusive activities the sequence of transactions are retrieved from the Data cache. The outcome of query is shown Fig. 3 below.

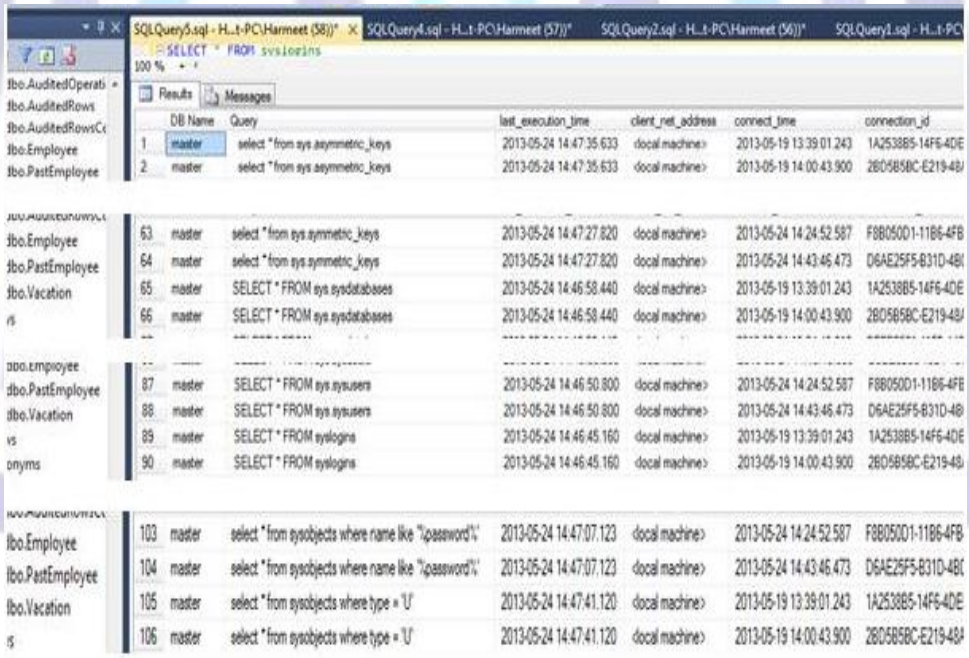

Fig 3: Reconnaissance acitivity retrieved from SQL server Data cache

The fig. 3 above shows the extraction from data cache showing results for attribute_ID_seq, DBname, Query, last_execution_time, client_net_address, connect time and connection_id. The user here is executing the system queries, queries to reveal passwords and the data encryption keys. The sequences are randomnly carried out which should raise the alarm.

To accomplish this the sequence of activity can be an effective way for representing user transactions and thus sequence alignment can be used to detect any anomalous activity. Sequence alignment is a technique used to quantify and evaluate similarity between two or more sequences. Any sequence alignment tool can be used for comparing database access patterns of genuine transactions and intruders transactions. As intruders are not entirely familiar with the normal database access patterns of legitimate users, they usually show some inter-transactional as well as intra-transactional deviation in their database access. Thus, most of the intrusive activities can be recognized through an analysis of past database access patterns. Each new transaction obtained from the artifact is passed through the RBU and the new attribute sequence is aligned with each of the normal transaction sequences. The degree of dissimilarity (ds) is determined based

on the dissimilarity between the new sequence and the user's normal transactions. A simple scoring system is used to evaluate the degree of dissimilarity. A unit match score δ (0 < δ ≤ 1) is assigned to each matched element and a unit mismatch score δ (0 < δ_ ≤ 1) to each mismatched element. Let 'L' be the length of the new sequence and 'M' be the number of matches with the aligned matched sequence. The degree of dissimilarity (ds) is then evaluated by the following equation (6):

$$d_s = \begin{cases} \dfrac{\delta'(L-M)-\delta M}{L} & \text{if } \delta'(L-M) > \delta M \\ 0 & otherwise \end{cases} \quad (6)$$

In this work, attribute_ID_seq is used as the transactional feature for sequence alignment deviation detection.

Spatio-Temporal Outlier Detection (R2):

Forensic analysis involves temporal detection, the determination of the time. Forensic analysis also involves spatial detection, the determination of "where," that is, the location in the database of the data altered in a database. (Note that the use of the adjective "spatial" does not refer to a spatial database, but rather where in the database the intrusion occurred). Similar transactions carried out by a user at certain location and time can be visualized as part of a cluster. Such observation is known as a spatio-temporal point which is scale-dependent on space and time. Normal spatio-temporal activity patterns of each user can be mined and used for detection of malicious activities in databases. Some deviation is normally seen from the normal transactions which can be detected as exceptions to the cluster. A spatio-temporal outlier (ST-outlier) can be defined as a spatio-temporal referenced object whose thematic attribute values are significantly different from those of other spatially and temporally referenced objects in its spatial and temporal neighborhood. In our work we have taken the example of SQL injection attack as sign of intrusion as shown below in Fig. 4.

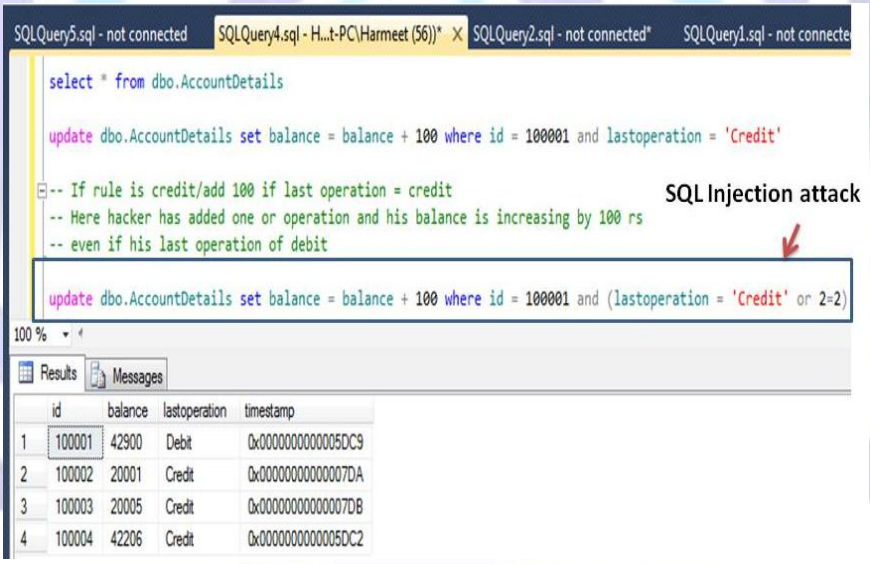

Fig 4: SQL Injection attack to update the column 'balance'

To show the intrusive activities the sequence of transactions are retrieved from the SQL Data cache which is shown below in fig 5.

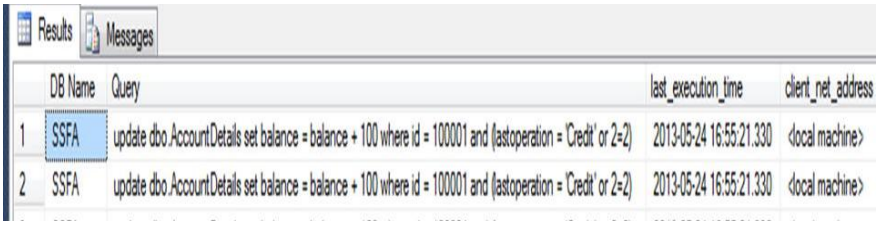

Fig 5: Traces of SQL Injection attack from SQL Data Cache

The Fig. 5 above shows the extraction from data cache showing results for DBName, Query executed, last_execution_ time, client_net_address

The application of database forensics with Dempster Shefer rule in this scenario would help an investigator to verify whether the attempted SQL injection attack was successful and, if so, which specific data was disclosed. To achieve this goal, an investigator could compare the attempted SQL Server injection attacks identified within logs and memory dumps against actual statements executed within SQL Server to verify whether the attack code was correctly tunneled and executed by the database. Taking the investigation a step further, the investigator could determine which, if any, data was added or changed since the time of the intrusion and rerun the T-SQL statements executed via the SQL injection attack to determine exactly which data (column) was tampered during the attack.

An approach is utilized based on the distance-based outlier (DB-outlier) detection technique proposed in [20], to filter out spatio-temporal outliers (ST-outliers). To verify the transaction, let N be the number of objects in the input dataset D and let DF be the underlying distance function that gives the distance between any pair of objects in D. An object O in a dataset D is considered to be a DB(p, d) outlier if at least a fraction p of the objects in D lie at a distance greater than d from O (d-neighborhood denoted by d_N). Let M represent the maximum number of data points within an outlier's d_N (i.e., M = N(1−p)). The clusters can be formed by using different attributes, the attributes (client_net_address,last_execution_time,table_ID_seq) are used for generating ST-outliers. Distance function DF is computed by applying Euclidean distance, which can be expressed as follows:

$$DF = \sqrt{(loc\_diff)^2 + (time\_diff)^2 + (tdist\_diff)^2} \tag{7}$$

Where loc diff : distance between current transaction location and the user's normal transaction location,

time diff : distance between current transaction time slot and the user's normal transaction time slot

tdist diff : schema distance between current transaction table ID seq and the user's normal transaction table ID sequence.

For computing tdist_diff, a distance measure is used similar to that suggested in [21]. A database schema S is assumed with a set RS of relation schemas. Attributes are structurally close if they belong to the same relation or can be related by exploiting a sequence of foreign key dependencies. Suppose two attributes $a_i \in r_1$, $a_j \in r_2$ where $r_1, r_2 \in RS$. The pairwise schema distance between a_i and a_j, denoted by PS_dist is defined as:

$$PS\_Dist(a_i, a_j) = \frac{SD(r_1, r_2)}{max\{SD(r_k, r_l) | r_k, r_l \in RS\}} \tag{8}$$

where $SD(r_1, r_2)$ is the shortest distance between r_1 and r_2. Given a set of attributes $A = \{a_1, a_2, ..., an\} \subseteq$ attributes(S), the schema distance function denoted by tdist_diff, is defined as:

$$tdist\_diff(a_1, ..., a_n) = avg\{PS\_dist(a_i, a_j)\} \tag{9}$$

We measure the extent of deviation of an incoming transaction by its degree of ST_outlierness. Suppose $DF_{avg}(T^{U_k}_{j,\rho})$ and $DF_{max}(T^{U_k}_{j,\rho})$ denote average distance and maximum distance of an outlier transaction $(T^{U_k}_{j,\rho})$ from the set of existing clusters in C' respectively. The degree of ST_outlierness (d_{STO}) of $(T^{U_k}_{j,\rho})$ is then given by

$$d_{STO} = \begin{cases} \dfrac{DF_{avg}(T^{U_k}_{j,\rho})}{DF_{max}(T^{U_k}_{j,\rho})} & \text{if } |d_N| \leq M \\ 0 & otherwise \end{cases} \tag{10}$$

Each of these rules R_1 and R_2 gives independent evidences about a transaction's behavior, which are combined by the Belief Combination Component. In this paper two specific techniques are used as rules, functionality of the RBC component can be further improved by incorporating new rules for identifying intrusion as per the system requirements.

2) Belief Combination Unit: The role of the BCU is to combine evidences from the rules R_1 and R_2 and compute an initial belief for each transaction traced. EDST provides a rule for computing the confidence measures of three states of knowledge: Intrusive (I), NOT Intrusive (I') and suspicious(unknown) based on the data from new as well as old evidence from the logs. Hence, we use EDST for combining evidences for this problem. The UD consists of two possible values for any suspected transaction $(T^{U_k}_{j,\rho})$ which is given as UD = {I, I'}. For this UD, the power set has three possible elements : hypothesis h = {I} implying that transaction $(T^{U_k}_{j,\rho})$ is Intrusive, hypothesis h' ={I'} that it isn't, and universe hypothesis UD that transaction $(T^{U_k}_{j,\rho})$ is suspicious.

The basic probability assignments (BPAs) for the two rules R_1 and R_2 can now be given as follows:

- BPA for R_1: For a transaction in which attrib_ID_seq does not match completely with the normal transaction attrib_ID_seq, we make the following basic probability assignments using the degree of dissimilarity (d_s) given by Eq. (11):

$$\begin{aligned} m_1(h) &= \frac{\delta'(L-M)-\delta M}{L} \\ m_1(\overline{h}) &= 0 \\ m_1(UD) &= 1-\left(\frac{\delta'(L-M)-\delta M}{L}\right) \end{aligned} \tag{11}$$

- BPA for R_2: For a transaction detected as an ST-outlier, the following basic probability assignments is made using the degree of ST_ outlierness (d_{STO}) given by Eq.

$$\begin{aligned} m_2(h) &= \frac{DF_{avg}(T_{j,\rho}^{U_k})}{DF_{max}(T_{j,\rho}^{U_k})} \\ m_2(\overline{h}) &= 0 \\ m_2(UD) &= 1-\left(\frac{DF_{avg}(T_{j,\rho}^{U_k})}{DF_{max}(T_{j,\rho}^{U_k})}\right) \end{aligned} \tag{12}$$

The zero in the BPA of h' in Eqs. (11) and (12) means that neither of the rules R_1 and R_2 gives any support to the belief that transaction $(T_{j,\rho}^{U_k})$ is genuine. Following Eq. (10), the combined belief of R_1 and R_2 in 'h' is expressed as:

$$P(h) = m_1(h) \oplus m_2(h) \tag{13}$$

Based on the initial belief P(h), a transaction can be initially classified as legitimate or suspicious.

The proposed mechanism for the detection of malicious database transactions has been depicted in the block diagram of Fig. 6 below.

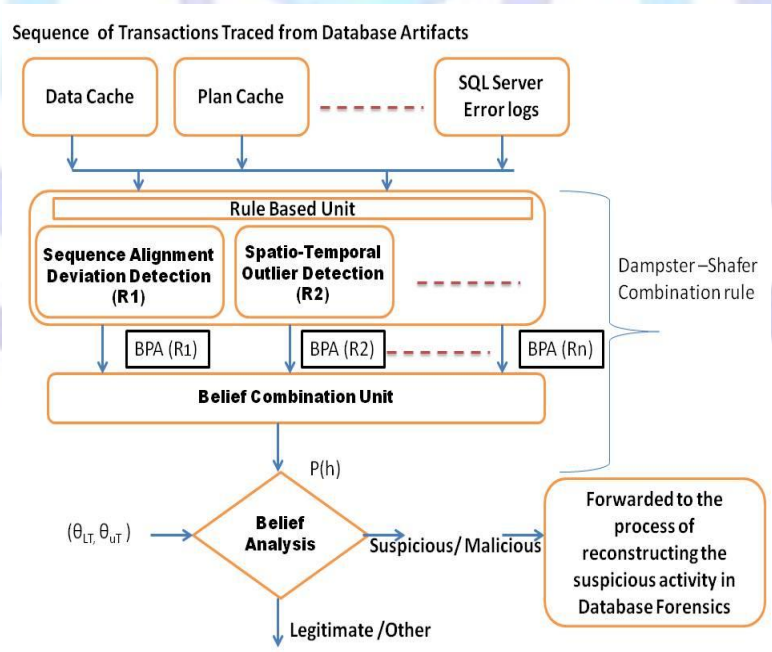

Fig 6: Block Diagram of the Proposed System

Each transaction traced from the SQL Errorlogs and cache is first analyzed by the Rule Based Unit of the system. However decision making occurs in two stages. In the first stage, the basic probability values BPA(R_1) and BPA(R_2) assigned by the RBU are combined using the BCU to get the initial belief P(h) for the transaction. If $P(h) < \theta_{LT}$, the transaction is considered to be genuine and is allowed to go through. On the other hand, if $P(h) > \theta_{UT}$ then the transaction is declared as malicious. In case $\theta_{LT} \le P(h) \le \theta_{UT}$, the transaction corresponding attribute_ID_seq is labeled as suspicious.

The suspicious transactions discovered will then be processed along with other evidences discovered from logs. This can be used for reconstructing the suspected activity.

CONCLUSION

Thus aim here is to allow you to distinguish suspicious transactions from several transactions among the relevant SQL Server artifacts during your investigation and eliminate nonrelevant database data. This ability to select and prioritize artifacts will reduce the amount of time you spend on data acquisition and analysis while ensuring that your database investigation remains manageable and achieves its objectives. The Database Server forensics during a digital investigation with Damster – Shafer Combination rule can prove or disprove the occurrence of a data security breach. It can **d**etermine the scope of a database intrusion or tampering within database. It can retrace user DML and DDL operations. This process can Identify data pre- and post-transactions. The retrace procedure from logs can recover previously deleted database data. In this paper, we have developed a approach for investigating intrusions in databases by collecting information/ transactions from various artifacts and use of belief update. The modified Dempster's rule is applied to combine multiple evidences from the rule-based unit for computation of initial belief about each transaction. While combining rules using extended Dempster-Shafer theory analyzes only suspicious transaction, thus saving time for further investigation process.

REFERENCES

[1] E. Bertino and R. Sandhu. 2005. Database Security − Concepts, Approaches, and Challenges. In IEEE Transactions on Dependable and Secure Computing, Vol. 2, No. 1, Pages 2-19.

[2] R. Richardson. 2011. CSI/FBI Computer Crime and Security Survey. www.gocsi.com

[3] Epsilon, Sony and X-Factor database hacks part of a cybercriminal strategy claims SecurEnvoy. 2011 http://www.infosecurity-magazine.com

[4] Sarbanes-Oxley (SOX) Compliance Checklist SOX-Compliance.2011.https://correlog.com/support-public/SOX-Compliance.pdf

[5] HIPAA, US Department of Health & Human Services 1996. The Health Insurance Portability and Accountability Act. (1996). https://www.cms.gov/Regulations-and-Guidance/HIPAA-Administrative- Simplification /HIPAAGenInfo/ /index.html (accessed in 2013).

[6] HIPAA Regulations. 2013. http://www.gpo.gov/fdsys/pkg/FR-2013-01-25/pdf/2013-01073.pdf

[7] Kyriacos E. Pavlou and Richard T. Snodgrass. 2008.Forensic analysis of database tampering, ACM Transactions on Database Systems (TODS) 33(4):Article 30, 47+25 pages.

[8] Nina Godbole and Sunit Belapure. 2011 Cyber Security, Understanding Computer Forensics and Legal Perspectives. Wiley-India. ISBN: 978-81-265-2179-1.

[9] Kyriacos E. Pavlou. Database Forensics in the Service of Information Accountability, SIGMOD/PODS PhD Poster Session,2011. Poster Presented.

[10] Kyriacos E. Pavlou and Richard T. Snodgrass. 2012 .Dragoon: An Information Accountability System for High-Performance Databases. Demonstration. International Conference on Data Engineering (ICDE).

[11] Martin S. Olivier. 2009. On metadata context in Database Forensics, Digital Investigation, Elsevier, www.sciencedirect.com, Volume 5, Issues 3-4, Pages 115-123.

[12] Shafer, Glenn; Dempster–Shafer theory, 2002

[13] Kari Sentz and Scott Ferson.2002, Combination of Evidence in Dempster–Shafer Theory, Sandia National Laboratories

[14] Dempster, A. P. 1967. Upper and lower probabilities induced by a multivalued mapping. The Annals of Mathematical Statistics 38 (2): 325–339. doi:10.1214/aoms/1177698950

[15] F. Campos and S. Cavalcante,2003 An Extended Approach for Dempster-Shafer Theory In Proceedings of the IEEE International Conference on Information Reuse and Integration, Pages 338-344.

[16] Peter Frühwirt, Markus Huber and Martin Mulazzani, Edgar R. Weippl.2010. InnoDB Database Forensics, 24th IEEE International Conference on Advanced Information Networking and Applications.

[17] Harmeet Kaur Khanuja and Dr. D. S. Adane. 2011. Database Security Threats and challenges in Database Forensic: A survey, Proceedings of 2011 International Conference on Advancements in Information Technology (AIT 2011), available at http://www.ipcsit.com/vol20/33-ICAIT2011-A4072.pdf

[18] Harmeet Kaur Khanuja and Dr. D. S. Adane. 2012. A Framework For Database Forensic Analysis. Published in Computer Science & Engineering: An International Journal (CSEIJ), Vol.2, No.3.

[19] Suvasini Panigrahi , Shamik Sural , A. K. Majumdar Detection of Intrusive Activity in Databases by Combining Multiple Evidences and Belief Update. 2009.. IEEE Symposium. Published in: Computational Intelligence in Cyber Security.

[20] E. M. Knorr, R. T. Ng and V. Tucakov. 2000. Distance-based Outliers: Algorithms and Applications. Published in International Journal on Very Large Data Bases, Vol. 8, Pages 237-253.

[21] C. Y. Chung, M. Gertz and K. Levitt . 1999. DEMIDS: A Misuse Detection System for Database Systems. In Proceedings of Integrity and Internal Control in Information System, Pages 159-178.

[22] Kevvie Fowler.2009. SQL Server Forensic Analysis,ISBN:9780321533203, Addison–Wesley.

Authors Profile

Harmeet Kaur Khanuja, currently a Researcher in the area of Database Forensics. She is working as Assistant Professor in the Department of Computer Engineering at Marathwada Mitra Mandal's College of Engineering. Pune, India. She is a life member of ISTE. She has presented and published papers in several International Conferences and Journals. Her areas of interest are Information security Applications, Digital Forensics and Mobile Computing.

Dr. D. S. Adane, currently is a Professor and Head of Information Technology Department at Ramdeobaba

College of Engineering and Management, Nagpur, India. He received Ph.D. in Computer Science and Engineering from VNIT Nagpur, India. He has over 15 research papers to his credit in reputed International

Journals / Conferences and also reviewed the papers for many. His research interests include Distributed and Mobile Computing, Mobile Agents and Network Security.