

## A Practical Approach in Detection and Prevention of Insider Multi Transaction Malicious Activity

Mojtaba Dashti<sup>1</sup>, Imran Ghani<sup>2</sup>, Leyla Roohi<sup>3</sup>

<sup>1,2,3</sup>Universiti Teknologi Malaysia(UTM)Malaysia

<sup>1</sup>Mdkseyyed2@live.utm.my, <sup>2</sup>imran@utm.my, <sup>3</sup>Rleyla3@live.utm.my

**Abstract:** Information is probably the most valuable asset of many people and information systems. However, some data cannot be memorized easily. As a result, these data and information should be stored in a special place to recall later. Several technical methods and process for keeping data was designed by technology improvement such as database which is a data collection in a special arrangement and structure. Data protection is not only essentially important for some users like business users but also for nonprofessional's computer users. Since some events such as natural disaster and human behavior causing harms and much more cost, evaluating possible threats and susceptibility of system which is employed to protect the data should be taken into account. Therefore, securing data by using some new methods is a significant subject. Many studies prove that the "insider threat" is the most dangerous information security threat in advanced technological organization. It can be inferred that workers employ their individual privileges to do actions on the basis of their information and have knowledge of some susceptibility of the system might be an insider and perform some attack. As a result, employee's attack has significant hazardous and critical effects. In this paper a practical approach in prevention and detection of such malicious attack is introduced.

[Mojtaba Dashti, Imran Ghani, Leyla Roohi **A Practical Approach in Detection and Prevention of Insider Multi Transaction Malicious Activity.** *Life Sci J* 2014;11(9):632-643]. (ISSN:1097-8135). <http://www.lifesciencesite.com>. 101

**Keywords:** Security, Insider, attack, database, malicious activity, mitigation

### Introduction

Everyone knows the significant of data which have special meaning to all of us and many people are involved with it in their daily life, such as using it in cost of products, account number, address of home, city post code etc. These examples show simple data to memorize, however some data cannot be memorized easily. As a result, these data and information should be stored in a special place to recall later. Different ways have used to reserve data, for example writing data on paper or engraving on rock which requiring a lot of time. Several technical methods and process for keeping data was designed by technology improvement such as database which is a data collection in a special arrangement and structure. Thus, it is an alternative providing opportunity to protect data.

Data protection is not only essentially important for some users like business users but also for nonprofessional's computer users. Since some events such as natural disaster and human behavior causing harms and much more cost, evaluating possible threats and susceptibility of system which is employed to protect the data should be taken into account. Therefore, securing data by using some new methods is a significant subject. In following, importance of data security is presented:

- (i) Protection from of unauthorized data observation
- (ii) Protection from unauthorized data modification

(iii) Security of the data confidential

(iv) Specific protection of integrity of data

(v) Verification of only data availability to authorized user

In order to preserve data with CIA (confidentiality, integrity and availability) deficiency which without them data can be destroyed or lost, database applications should be used. Thus, some novel techniques such as user authentication, user privileges, data encryption, auditing (Rathod *et al.*, 2012), and etc. are suggested to protect data from mentioned damages. There are different types of attacks on database that is describe in Chapter 2; however intruder attack is a significant attack among them. Intruder attack includes two types' attacks named insider and outsider, whereas outsiders attack is more prevalent, so lots of Intrusion Detection Systems (IDSs) are designated to protect from it (Heights, 2009). To sum up, while a larger portion of attacks includes outsider attacks, but the insider tacks are more severe (Mun *et al.*, 2008).

Many studies prove that the "insider threat" is the most dangerous information security threat in advanced technological organization. The dangerous manner of user about database is principally the meaning of insider attack or it can be defined as "A person who has privileges to access the underlying system" (Nguyen *et al.*, 2003). In addition, insider is an individual who has the knowledge of the organizations information system structure to which he/she has authorized access" (a particular person knows the structure of organizations

information system which has permission legally to access” is described as an insider). From (Maybury *et al.*, 2005) it can be inferred that workers employ their individual privileges to do actions on the basis of their information and have knowledge of some susceptibility of the system might be an insider and perform some attack. As a result, employee’s attack has significant hazardous and critical effects (Nithyanandam *et al.*, 2012)..

### Database Security

lots of main parts of information, that is critical to organizations is supervised by Database system since all accesses without authorization or all forbidden database managements can cause exception of problem for the organization. People who are not permitted to entry as well particular sections of the database or all parts the database should be prevented. This means that the considerable challenge is security of database against approaches or operation lack of authorization.

The greatest significant issue, which organizations encounter during execution a database system, in spite of lowest concerning of it, is the data protection or security. The subsequent subjects are protected by Database security.

**(i) Database Privacy:** entry or access to the database has to be permitted just authorized persons. Moreover, just the section of the database which is needed for the functions have to be accessible to them and this means that users are authorized to entry just the data which is related to their works.

**(ii) Database integrity:** To sustain database integrity, database have to be secured from incorrect changes, whether it is planned or not. In fact, Just performances which require to be done through the user should be permitted to them. Workers, For instance who are not members of accounts department have not be permitted altering the balance sheet of the organization. Therefore Just that department’s workers have to be permitted to do.

**(iii) Database availability:** The permitted users to execute their activities on the section of the database which is accessible to them have not been confined by security. The balance sheet, as an example has to be updated via an accounts department worker.

The application of an extensive range of information security management or controls to secure databases opposite compromises of their secrecy or confidentiality, unity or integrity and accessibility is taken into account by database security. Furthermore, databases in that system possibly consist of the data, (database applications or stored functions) the database functions or saved applications, the database systems, the database servers and the connected network links. Technical, procedural/administrative and physical are respective samples of different kinds or classes which are involved in information security controls or

managements. A professional subject inside the extensive region of computer protection, information protection, danger management (computer security, information security and risk management) is a database security.

Security risks to database systems include:

(i) illegitimate or unintentional action or abuse by permitted database users, network/systems managers or database administrators by illegitimate users or hackers intruders ( for example, improper entry to vulnerable data, metadata or operations inside databases, or improper modification to the database programs, structures or security configurations);

(ii) Malware (malicious-logic infections) leading to happen some events like unpermitted entry, data exposure or individual leak of data or proprietary data leakage, obliteration of or harm programs or the data, disruption or refusal to accept permitted entry to the database, attacks on other systems and the unpredicted inability of database services;

(iii) Load beyond capacity or overloads, function restriction and volume or function or capacity problems of capacity issues causing the permitted users being incapable to employ databases deliberately;

(iv) Physical harm to servers of database is resulted due to lightening, floods, becoming too hot, stationary unloading, spilling of liquid unintentionally, collapses, destructions and obsolescence of electronic devices and fires in room of computer;

(v) Design flaws and programming bugs in databases and the connected systems and programs, producing different security susceptibility (as an example, unpermitted benefit increase), destruction of data or data corruption, execution reduction and so on;

(vi) Immorality of data and/or data failure resulting through the access of expired data or controls, in system or database administration procedure’s errors, maliciously sabotage / illegal harm and so on.

data leakage, obliteration of or harm programs or the data, disruption or refusal to accept permitted entry to the database, attacks on other systems and the unpredicted inability of database services;

(iii) load beyond capacity or overloads, function restriction and volume or function or capacity problems of capacity issues causing the permitted users being incapable to employ databases deliberately;

(iv) Physical harm to servers of database is resulted due to lightening, floods, becoming too hot, stationary unloading, spilling of liquid unintentionally, collapses, destructions and obsolescence of electronic devices and fires in room of computer;

(v) Design flaws and programming bugs in databases and the connected systems and programs, producing different security susceptibility (as an example, unpermitted benefit increase), destruction of data or data corruption, execution reduction and so on;

(vi) Immorality of data and/or data failure resulting through the access of expired data or controls, in system or database administration procedure's errors, maliciously sabotage / illegal harm and so on.

Numerous information security control's layers and types are allotted to databases which is comprising of:

- (i) Auditing
- (ii) Access control
- (iii) Encryption
- (iv) Integrity controls
- (v) Application security
- (vi) Backups
- (vii) Authentication

#### Attacks On Database

Two various types of risks to security of database can be listed which are logical and physical threats. Firstly, logical threats which have not permitted logical reasonable entry to information, generally by way of software may cause denial refusal to accept of service, information disclosure and data modification change. Secondly, some of the physical threats are composed of (however are not restricted to) storage devices ruin, compulsory passwords disclosure, inabilities of power, and robbery. In order to impede this kind of risk the greatest usual method is to restrict the access to the storage devices, copy of data in order for put backup in case of a malfunction and recovery processes situated properly.

(i) **Insider Threat:** for a database a corrupt authorized is one of the most, important threats which is able to legitimately access confidential information and afterwards in an electronic manner or by some other methods like word of mouth or printout this information possibly would be leaked. To impede it from the database management system just few actions can be done. Through the preventing a user to log in with systematically classified access to save or copy the data to a place with unclassified access, obligatory access controls can assist slightly. In addition, through the restricting the users' amount accompanied by access level and differently complex processes normally can manage this kind of danger (Parveen *et al.*, 2011).

(ii) **Login Attacks:** Another way to compromise a database is to successfully log in as a legitimate user. This can be done by physically stealing the information or monitoring network traffic for login information. Another attack could involve accessing password lists stored in an operating system. Login information can only be as secure as the password used. If it is easy to crack, there is not much that can be done. Restrictions on the type and form of passwords can help, but does not solve the problem. The database must employ authentication and encryption to ensure that this type of attack is less likely. Different method in order to prosperous log in as an authorized user is

compromising a database that can be performed by information theft in a physical manner or supervision network traffic to sign onto computer system for information. moreover, another attack are able to cause gain entrance to password lists collected in an operating system and login information certainly can just be protected similar to the password used which also is protected. Very few actions can be performed on the conditions that crack can without great difficulty be happen. And also, it would be impossible to solve the problem even some methods such as restrictions on the type and form of passwords are done. Authentication and encryption have to be used by the database to make secure from this sort of attack. The web server could be established set up in two different ways to approve pass the user authentication information straight to the database or as well as confirm authenticate the user and after that employ the authentication information of web server to login to the database. The user confirmation authenticate which is mentioned second way, supplies maximum improvement of efficiency optimization since the connection is able to be conceal cached that cause still greater unprotected system since in the case of the web server is compromised therefore the database is too compromised. A one-time password system or Encryption can be employed usually. Moreover, an administrator has potential to watch network traffic to obtain login information of worker in our EERSS sample implementation application and administrator can modify the data of worker like rebuke or dismiss the worker.

(iii) **Network Attacks:** Many attacks might be happening on a database on the condition that it is accessible over a network, and if the network is the Internet attacks possibility will be increased. A firewall, one of the safety measures could be situated properly to secure the database and maybe the Web server. By several methods can secure the data which sent over the network. For example the secure socket layer is a common method on the Internet which can impede an attacker from only collecting information through observation network traffic. In addition, there would be requirement to a proper procedure to authenticate with the database. In addition, to secure authentication in connection with databases, certificates can be employed. The refusal to denial of service attack is a particularly usual attack and this sort of attack is more connected to the Web server which permitting access to the database, however, it is able to rise against a database itself.

(iv) **Inference Control:** User of a database can exploit information since they can entry to and perhaps somewhat additional external information to derive information that they are not able to access. In order to describe in detail, information at a maximum level of protection can be inferred from information at a

minimum level of protection which has an extremely troublesome risk to hinder. Moreover, this risk is related to databases based on statistical databases typically. Through responds to authorized statistical questions on the information, it is possible to infer data about particular persons. An unsophisticated and overly simplified access is able to transfer the information with lower level to a higher level data.

In order to prevent the speculation inference just the lowest level data is required which have to be transferred to the higher security level. In spite of the fact that this usually causes more than, if not, whole the lower level data is going to re-categorize at a higher security level, however, it is not normally a satisfactory solution. Query restriction, data perturbation, and output perturbation are some other methods which can be employed. Query restriction demand to a least number of rows to comprises a piece of the query and in fact it cannot resolve the problem, however it raises the amount of queries which is essential to conclude any confidential information. Verifying an account auditing that query samples in desire that such action will be discovered proceeding compromising the confidential information, is a usual solution.

(v) **Trojan Horses:** A kind of Immoral Corrupt software implementations applications causing leakage of confidential information are defined as Trojan horses which are employed regularly in a system, however in order to duplicate or transmit important information to users or locations without authorization, have been changed. Installation on the system is essential for applications which have Trojan Horse and attackers or administrators who do not understand the existence of Trojan Horse in the application can install it. As awaited the immoral application will activate for whole applicable aims. However, it will be acting several extra unlawful operations too. Each worker, for instance can make an application having a Trojan Horse and the worker can cause the administrator to apply this application in his weekday work. As a result, this application is duplicating whole the data which just the manager has access to their locations in the database and now the worker can access to them since of Unaware manager. Thus the worker can access to data of other workers which he couldn't have approach to. This attack can be prevented if obligatory access controls substitute for discretionary. Through this access control, the worker cannot approach the sorted information classified data since the manager's clearance level would not let inscribing them to a lower security level.

#### **Insider Threat**

Insider threat is a sort of threat which is usually consider as related to authorized users and these users influence their system benefit privileges malevolently, freedom familiarity and vicinity proximity to their

computational environment in order to impose harm or compromise important data. A meaningful part of reported events is created through internal attacks and insider misuse which is according to the annual CSI/FBI surveys directed from 2005. Insider attack is still the most powerful indication which the recent study directed by CERT and the US Secret Service has shown that. In a real world environment, US Secret Service is the first group which supplies a complete understanding of the problem. But any of understood part of process cannot efficiently solve this problem. Comprehending the threat as an initial difficulty should be considered (Chinchani *et al.*, 2005).

#### **(i) Insider Threat is a Low Base Rate Problem.**

It is hard to prophesy or defend against insiders attacks since criminal perpetrators of insiders attacks are users having legal permission. As a result these inevitable attacks causing in inactiveness are observed through security officers (Chinchani *et al.*, 2005).

(ii) **Insider Threat is Mis-perceived.** In many cases organizations uniquely focus on external attacks principally due to the fact that protection verify accounts security audit devices and modeling methods are quickly and easily attainable which assist in discovery susceptibilities and repairing them. On the other side, since its measurement is hard so insider threat is not discerned properly and because of devices and methods shortage hence the condition does not improved. It is an important progress if any valid template model or estimation assessment methodology is suggested (Chinchani *et al.*, 2005).

(iii) **Insider Threat is High Impact.** Insider attacks possess a greater ratio of accomplishment and they are able to remain undetected and have higher danger than external attacks in spite of the fact that insider attacks cannot happen as regularly as external attacks. Also as a result of insiders get specific valuable benefits over exterior enemies external adversaries it is occurred. They are common about their aims and the security countermeasures in place. Hence, just with a short or non-existent survey reconnaissance phase stage, extremely deleterious attacks are able to be activated (Chinchani *et al.*, 2005).

While all these practical measures will reduce the threat, they cannot eliminate it altogether, and some incidents can still occur. A possible solution it would seem is an overall increase in monitoring, logging and security countermeasures. However, it only leads to general inconvenience. Moreover, in an organization, it sends wrong signals of distrust between the management and the employees. In continuous, some methodology and model to assumption, presentation, and detection insider threat are illustrated.

In summary, both computational main components and human factors are involved in insider threat which is a complicated problem. To make less

this threat Stages like instruction and learning and filtering of workers in advance employment can be taken on as a long period procedure. They cannot remove it completely, and some events are able to yet happen, although the risk will be decreased through whole these applicable steps. Generally intensify in supervision; logging and security countermeasures would appear to be a probable solution, nevertheless, it just directs to usual disturbance. In addition, in organizations, mistake signals of mistrust would be transmitted between the supervisors and the workers. Here is demonstrated several principles, methods and model to hypothesis, submission, and detection insider threat in the following.

#### Insider Threat Detection Methods and Concept

Within this section, insider threat detection models on database are explained. Different techniques are applied to describe the proposed models.

The first papers that is review, are the research of Yip Chung in 2000 (Chung *et al.*, 2000). In this research, the author proposed a misuse detection system tailored to relational database systems. This system which is known as DEMIDS (Detection of Wrong use in Information resource Systems) provides a pure and useful set of tools to acquire user profiles from audit logs. The mechanism of these profiles are to specify the typical amount of features that are audited in audit logs, hence they prescribe the typical behavior (access patterns) of clients as well as to identify misuse behavior.

Although it can be used to detect intrusion and insider abuse, DEMIDS places focus on the recognition of malicious behaviour by genuine customers who strive to take advantage of their authorities. Therefore, the techniques are especially useful for internal control. This system is capable of complement misuse recognition at the operating system layer because intrusion affairs that MDSs don't succeed to recognize at the operating system layer may be detected as uncommon event at the database system layer. Moreover, the extracted profiles can serve as a worthwhile means of security re-engineering of a company by assisting the security officer (SSO) to define/refine security policies and additionally to confirm current security policies, if there are any. Lastly, profiles possess the potential and the possibility of being applied to respective enforcing systems in the database systems using, e.g., triggers, assignment of privileges, or roles.

A necessary notification regarding this suggested method is that, given a database schema and associated programs, the accessibility patterns of individual users will give rise to structure working scopes composed of certain sets of attributes that are usually referenced together with some values. The concept of working scopes is well inspired by the concept of frequent item

sets which are sets of aspects with certain values. DEMIDS describes a view of distance measure based on the data structure and semantics (integrity constraints) protected in the data dictionary and the individual behaviour demonstrated in the audit logs; which appraises the nearness of a set of attributes with regards to the working scopes. Distance measures are used to lead the search towards frequent item sets in the audit logs by a brilliant data mining strategy that uses the efficient data processing ability of database management system. Any Misuse activities and crimes, such as tampering with the thoroughness of data, can be tracked, detected and rectified by evaluating the profiles against the protection guidelines and security policies specified within the organization or against new information (audit data) collected about the individual customers (Chung *et al.*, 2000).

The proposed misuse detection system DEMIDS is comparatively paired to a current database program where DEMIDS implements certain performance functionality of the system such as auditing and query processing. DEMIDS includes four elements (Figure 1): (1) Auditor (2) Data Processor (3) Profiler and (4) detector..

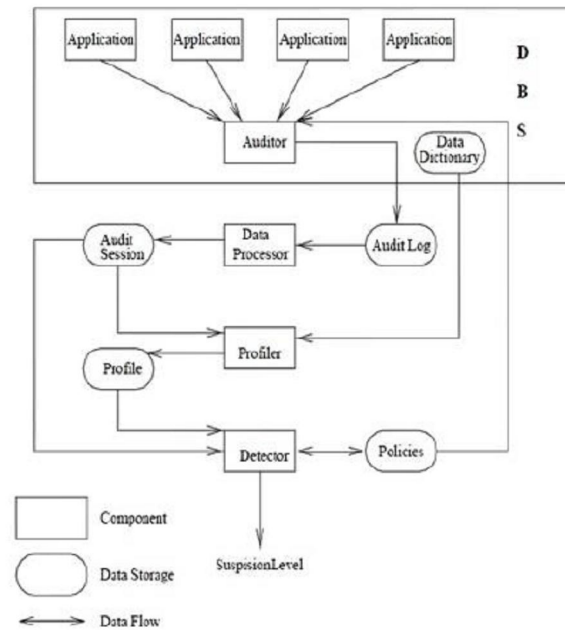
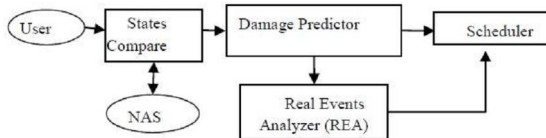


Figure 1: Components of DEMIDS's architecture (Chung *et al.*, 2000)

Additionally, latest version of methods focuses transactions that insert malicious attacks into the system and effect on other transactions by damaging them, where same goes to the data bases. The risk minimization and threat mitigation technique provided in this research puts much effort to eliminate the malicious impacts of an insider and to furnish

minimization service. The contents of this research are summarized as follows and its contribution is highlighted as below. First, Normal Activity Rules are established to dictate limitation on the relationship between data items and transactions that use different security levels. Second, a Predictive Dependency Graph is developed to examine and analyse various patterns of information movement flow amongst the data items. Third, to predict and cease malicious data circulation flow, the labelling mechanism is implemented to highlight any unverified data items. Fourth, the notion of a Real Event Analyser (REA) is applied in the work to check and monitor the transactions' effect actions pertaining to real world. The advantages attributed to REA is that following using the REA, tracking of the dubious users' actions and potential damage to the data source can be derived. Here we explain these systems one by one. (Li *et al.*, 2012)

Figure 2 reveals the structure of the suggested approach. It consists of two major processing parts: the Damage Predictor and the Real Event Analyzer. Transactions are first compared with Normal Activities States (NAS) and categorized into PNT and PTT transactions for simplifying recognition process in near future. The Damage Predictor processes transactions and subsequently produces Predictive Dependency Graph to estimate probabilistic damage and to report the alerts to the real event Analyzer and the Scheduler. The Actual event Analyzer (REA) investigates the implemented transaction and confirms the data items, and then reports back to the scheduler. (Li *et al.*, 2012)



**Figure 2:** System architecture (Li *et al.*, 2012)

Eventually, is proceeded to demonstrate and articulate the method suggested by Nahla Shatnawi in 2011. An insider threat model that covers such threats on the application level is built in this research. In this model, they strive to prevent any threats subject to insider and meanwhile discovering possible threats for individual tasks. As an indicator of the potency of our proposed model, we want to mitigate the false alarms percentage. In this model, the assumption of that the variety of tasks an insider can perform is restricted, has taken place. This means that the sequence of executing a transaction process can be defined in advance and therefore, the sequence of performing the associated transaction that are required to perform the associated task can be predefined too(Shatnawi *et al.*, 2011).

Within this model, they attempt to control and monitor the insiders by monitoring their nexus of

activities. Before implementing his/her activity, an insider shall to announce his/her intention by specifying the work that he/she needs to do. Based on that, a record of objects/ transactions that he/she needs to follow to complete his/her task and the data items that he/she can use will be furnished by the system. Subsequent to enumerating the set of components and transactions by the program according to the final aim of the user, this is turn to set up a dependency graph which shows dependency connections among components and/or transactions. This procedure can be conducted because we assume that the variety of tasks and the variety of transactions are restricted and predefined. According to these dependencies, the nexus of components to be used within a transaction and the nexus of the transactions themselves per se to be incorporated can be established.

Afterwards we articulate the details and operation features of this model. This model functions in two phases.

At the first phase, it controls the quantity of malicious activities that the user can do; and within the second level detects possible threats. The aim of the first step is fulfilled by posing limitations on the individual work and on the way of doing this work by the individual. In this level, the assumption is that the person work is limited i.e. the person can perform a predefined set of tasks. Each task possesses the potential of performing the work by using a little wide range of transactions in any particular order. at the second level where possible threats are detected, initially the user is obliged to identify his/her intended work which he/she wants perform under any particular application within the proposed model, meanwhile necessary to mention that the task must be amongst the list of tasks which he/she has got the authority to perform. Afterwards, once the user performs the process of task to be done, the system will immediately probe throughout the database; to find out and present the expected transactions and objects within a predefined set of transactions and objects for the ordered task the individual, also the existing similar transactions and objects for the relative tasks, if are available. The individuals using the transactions, is indeed using those tools (of the task or of the equivalent) in order to perform his/her own intended work; either in normal ways or as a threat.

#### Classification of Related Methods

As is illustrate in Table 1, it can be concluded that the techniques and their limitations which are debated above consist "Prevent and detect" protocol and they execute in various level. The greatest prominent inability that they have among several weaknesses, although is their incapability to detect multi transaction malicious transactions.

The following table reveals some techniques and their limitations which are considered previously. In these techniques, the process of “Prevent and detect” is taken into account and they present various levels

(grades or ranks). Several inabilities however can be listed which the predominant one is their inefficiency to detect (discover or discern) multi transaction malicious transactions.

Table 1: Comparing Methods

Article	Method Type	Type and Year	Solution	Limitation
Insider Threat in Database Systems: Preventing Malicious Users' Activities in Databases (Heights, 2009)	Prevention	2009 Sixth International Conference on Information Technology: New Generations	on the basis of user tasks roles Maps to some transactions	Inability of User to change any of the transaction in any application
DEMIDS: A Misuse Detection System for Database Systems (Chung et al., 2000)	Detection	Conference 2000: Department of Computer Science, University of California	employ audit logs to derive profiles Detect at the operating system level	no supportive insider multi transaction detection in application level
Malicious Users' Transactions: Tackling Insider Threat (Li et al., 2012)	Detection	Journal 2012	Set up several rules put to oblige the connection between data objects and transactions	Transaction definition in advance just influence on individual transaction
Detection of Insider Misuse in Database Systems	Detection	International multi conferences of engineering and computer science	Detection perhaps threat for individual role Supervise the insiders' through observation their activities progression of.	Insider roles is restricted The insider must to proclaim its aim preceding execution.

As content of table shows there are four works that we can categorize them in group of related works. In 2000 an approach was proposed by Christina Yip Chung under title of DEMIDS: A Misuse Detection System for Database Systems that turns researcher's attention to new area. After years, this work still is used to provide background and fundamental knowledge for new research in this area. The next related work refers to Insider Threat in Database Systems: Preventing Malicious Users' Activities in Databases that was proposed by Yi Hu in 2009. This approach presented a prevention model that based on user task, but this model did not contain detection. The third one that is also the most similar works to our proposed approach was in research topic of Detection of Insiders Misuse in Database Systems that was proposed by Nahla Shatnawi in 2011. This model proposed a detection approach to detect multi transaction malicious activity in database. This method tries to detect malicious task by following their sequence of transactions. The last and most recent approach that we focused in this research addresses a research that was conducted in

2012 by Weihan Li and title of Malicious Users' Transactions: Tackling Insider Threat that proposed a predictive dependency graph to determine a data flow pattern among data item. This model also follows user activity to detect threat on database.

#### **Design Of Prevention And Detection Malicious Transaction**

The proposed algorithm in this paper contains these main components. Actually this algorithm consists of combining two existing algorithms that have some limitations. The first algorithm, which users must define the transaction before executing is introduced by (Shatnawi et al., 2011) is a limitation that is solved by adding process two. Moreover process two is similar a transaction that introduce by (Chen, Lu, and Xie, 2007) that use log file to make an audit table. New things which are added to both of processes are two data storages that are policy and black list to have a faster detection and prevention. PDMT algorithm includes two processes that each process has some component. These components held in common between these two processes. Just detection process has one more

component which is use in this process that called log file. In continuous these components are introduced. Figure 3 shows components of algorithm.

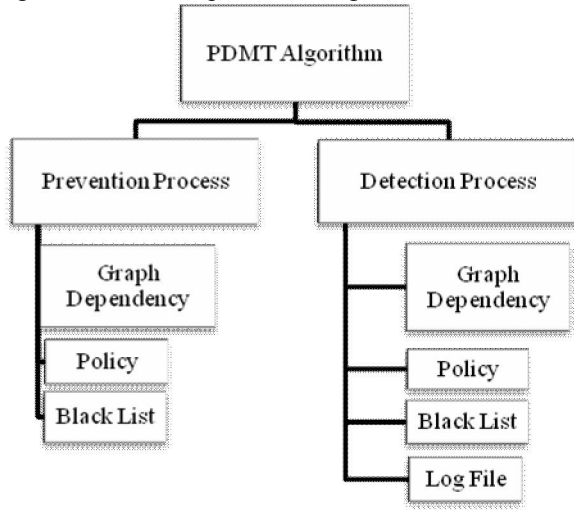


Figure 3: Algorithm Components

In the below main components of algorithm are described:

(i) **Main Algorithm:** The algorithm contains two main processes which called prevention process and detect process. Also method includes two data storage that help method to work as well to detect malicious transaction.

(ii) **Prevention Process:** This process work on users or application that are defined their transactions before executing.

(iii) **Detection Process:** This process work on users and application that are not define their transactions. Detection process monitor transaction during execute transaction by reading log file.

(iv) **Black List:** this data storage uses in both of processes. The method check transaction before execute with this data storage. Figure 4 show black list data storage in application.

Username	commandType	targetobject	filedetable	Tab
alokbar ahmadi	Select Select	tbl_lesson tbl_student	l_name,l_unit,l_s_name,family,s_birthday,	1
alokbar ahmadi	Select Select Update	tbl_field tbl_lecturer tbl_lesson	F_id,F_name, l_id,l_name, _birthday, l_id,	2
reza	Select	tbl_lesson_student lecturer	id_student,	1
maz	Select Select Select	tbl_lesson tbl_lecturer tbl_student	l_unit, l_name, s_name,family,	1
ds	Select Select	tbl_lesson tbl_student	l_name,l_unit,l_s_name,family,s_birthday,	1
hr	Select Select Update	tbl_field tbl_lecturer tbl_lesson	F_id,F_name, l_id,l_name, _birthday, l_id,	1
dashfi	Select Select	tbl_lesson tbl_lecturer	l_id,l_name,l_unit,l_name, _birthday,	1
mk	Select Select	tbl_lesson tbl_student	l_name,l_unit,l_s_name,family,s_birthday,	1
reza	Select Select Update	tbl_field tbl_lecturer tbl_lesson	F_id,F_name, l_id,l_name, _birthday, l_id,	2
reza	Select	tbl_lesson_student lecturer	id_student,	3
maz	Select Select Select	tbl_lesson tbl_lecturer tbl_student	l_unit, l_name, s_name,family,	2
ah	Select Select	tbl_lesson tbl_student	l_name,l_unit,l_s_name,family,s_birthday,	1
alokbar ahmadi	Select Select Update	tbl_field tbl_lecturer tbl_lesson	F_id,F_name, l_id,l_name, _birthday, l_id,	3

Figure 4: Black List Data Storage

(v) **Policy file:** This data storage contains rules and some privacy. Users and application must be following these rules. Most of the times it is almost impossible to prevent a system from malicious insiders and this kind of attack is very important when considering security in the field of databases. Fortunately, there is way to determine such activity; looking at real world cases of insider activity. It makes it possible to determine common patterns that show which controls would be the most effective and reducing the threat. This approach is very important as this can mitigate insiders attack.

information are vital to any company and by understanding insider risk and adopting sensible information security policies, the risk threats will be greatly decreased, either accidental or malicious ones.

Understanding and documentation of the employees who have access to which systems and information is very crucial in all these policies. It would be difficult or impossible to trace the privileges of the user to perform these tasks without such documentation. Finally, it is worth to talk more about the most critical administrative security policy control. Every organization has better to do documentation for the information security responsibility and access of key job roles. The policy is the basic part of the other parts of the organization, and enables the protection of information throughout the employment life cycle.

- Each user has to have user name and password.

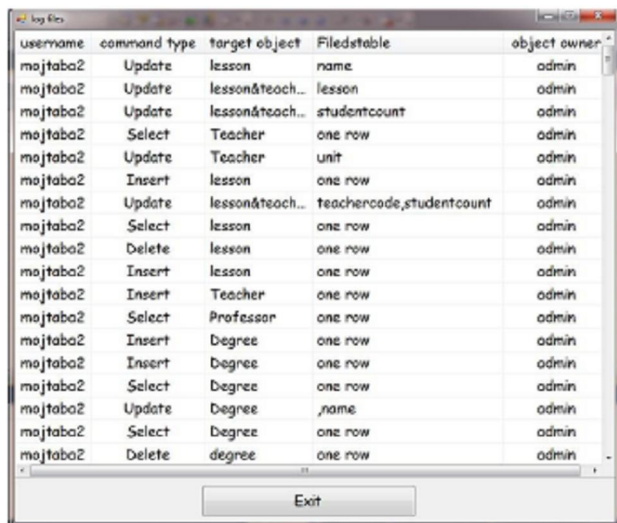


- Each user activity has to have submitted in log file.
- If an order of transactions in task is malicious, any order of these transactions is malicious.
- If user has permission to modify a data of table but has not permission to delete on this table, user can not modify all fields of this table.
- It is not acceptable if user wants to access to tables that are created a table which has not permission to access this table.

(vi) **Graph dependency:** In some fields such as computer science, mathematics and digital electronics a dependency graph defines as a directed graph which depict many items' dependencies in direction of each other. It might be able to extract an evaluation order or an evaluation order's absence which by using the dependency graph esteems the provided dependencies. In this method dependency graph uses for relation between tables and then compare with simulated transactions.

(vii) **Log file:** A DBMS uses a transaction log to keep track of all transactions that updates the database. The information stored in this log is used by DBMS for a recovery requirement triggered by 'Roll Back' statement.

But in this method log file uses for monitoring database to detect malicious transaction. Figure 5 show log file table in application.



username	command	type	target object	Filedstable	object owner
majtaba2	Update		lesson	name	admin
majtaba2	Update		lesson&teach...	lesson	admin
majtaba2	Update		lesson&teach...	studentcount	admin
majtaba2	Select		Teacher	one row	admin
majtaba2	Update		Teacher	unit	admin
majtaba2	Insert		lesson	one row	admin
majtaba2	Update		lesson&teach...	teachercode,studentcount	admin
majtaba2	Select		lesson	one row	admin
majtaba2	Delete		lesson	one row	admin
majtaba2	Insert		lesson	one row	admin
majtaba2	Insert		Teacher	one row	admin
majtaba2	Select		Professor	one row	admin
majtaba2	Insert		Degree	one row	admin
majtaba2	Insert		Degree	one row	admin
majtaba2	Select		Degree	one row	admin
majtaba2	Update		Degree	,name	admin
majtaba2	Select		Degree	one row	admin
majtaba2	Delete		degree	one row	admin

Figure 5: Log File

### Algorithm Architecture

Main implementation of PDMT method consists of two processes. These processes are called prevention and detection process. In this implementation, some requirements are needed. First is to define policy, which sequentially leads to define some rules and define some

limitations for relational database. Users and applications must follow these rules. Each transaction which wants to be executed and breaks the rules will be prevented by the mitigation process of the algorithm which may terminates that specific transaction and does not allow it to access database. As mentioned before, this method removes limitation of previous methods. One of this limitations is that a user or application have to define their transaction before execute. But in this method by eliminating this limitation and by adding a new process, users and applications which are not defined their transaction can execute their tasks as well. This process is called detection process. Also the first one called prevention process.

This method is worked between database and applications. Actually as mention before this method include two processes. The first process, prevention process work before executing transaction on database but the second process, detection process is work when a transaction is executing. Detection process works while transaction is running as parallel. Figure 6 shows a schema of algorithm.

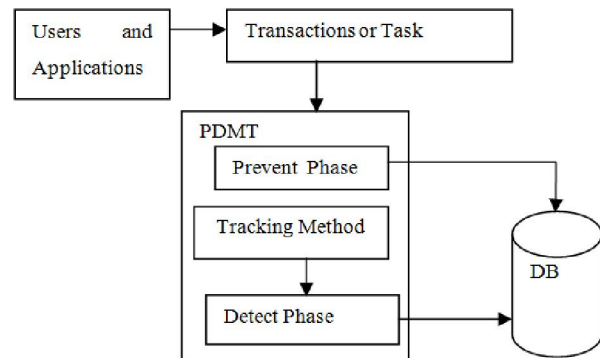


Figure 6: Components of PDMT's architecture

This algorithm execute in two main processes. Actually, these two process running separately. In below is shows process of algorithm.

(i) **Prevention Process:** works on transaction before execute transaction on database. In this process algorithm users must defined transactions and tasks before executing on database. Then algorithm analyzes the transactions.

(ii) **Detection Process:** works on transaction during the transaction is executing on database. The users and applications do not define transactions and tasks. The process check log file and then analyze the transaction.

**Algorithm Implementation** C# programming language will be used in this thesis in order to implement the specified algorithm and to evaluate the process parameters and enhancements. C# is a programming language for technical computing from Microsoft. Company and ideal for simulation

applications like the one is described in this thesis. C# allows designing attacks to database, designing functions of mitigation and prevention of attacks and other malicious activities to database. Also to save data and test algorithm is needed to database software which Sql server 2008 is selected..

**Prevention Process**

In this process, applications and users, which defined their transactions, are under process. Only authorized users and applications can execute their transactions on the system. The below steps introduce algorithm how to work.

(i) **Step1: Input Transaction:** In this step just authorized users and application has permission to access to database and execute transaction. User input username and password and then algorithm compare with existing username and password and then allow entering the system and access to data, which has permission. Then user defines task and transaction to executing.

(ii) **Step2: Identify Transactions:** In this step defined transactions analyze by the algorithm make an audit table to use analyze transaction. Information of transaction extracts by algorithm and saves in a table, which is called Audit table. This information includes table name, operation, field name, and user name.

(iii) **Step3: Check with Black List:** the transactions compare with a data storage which called

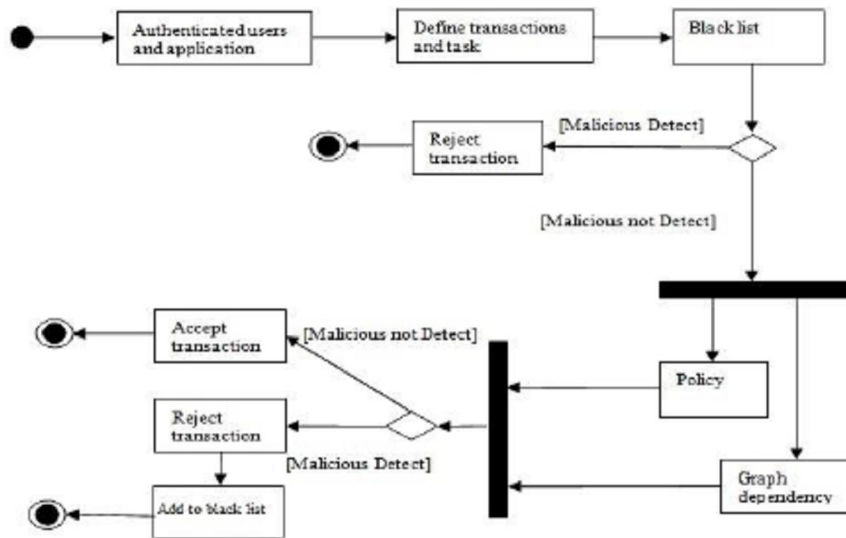
black list. This data storage is created by algorithm during executing detection process (process two) to have a faster search and prevent malicious transaction. Data that saved in black list are query, operation, tables, filed, and username.

(iv) **Step4:** Any misuse behavior find, algorithm prevent transaction to execute and reject it. Else algorithm run next step. In this step, rows in audit table compare with black list and if any same row find, algorithm prevent transaction to execute.

(v) **Step5: Policy Check:** after identify transaction algorithm check and compare simulated transaction with policy data storage that defines by database administrator.

(vi) **Step6.Graph Dependency:** check: like step six, algorithm compares simulated transaction with this data storage. Graph dependency show relationship between tables in database. in this table audit table compare with graph dependency table, if fields in transaction that are saved in audit table is equal with fields in graph dependency then check table name in graph dependency. If user has not permission to access this table, algorithm prevent execute transaction.

(vii) **Step7. Final check:** compression in two last steps check in final step. If any malicious find in two last steps, algorithm reject transaction, else accept transaction and allow execute it.



**Figure 7:** Prevention Process.

**Detection Process**

The second process, detection process is described in this section. After prevention process, detection process is running in order to monitor and trace transactions or tasks. The user and application that are not defining their transactions or tasks are monitored

and traced by detection process. When a transaction is executed, all activities are logged and saved in log file. The detection process tracks the user activity by reading activity log (log file) and analyzes it step by step. Then after each activity, method compares these three data storage. If any malicious detected, method will then

terminate transaction and roll back database to previous state. Finally tasks and transactions are added to black list if not exists. The below steps are show the algorithm mechanism.

(i) **Step1: start mode:** In this step just authorized users and application has permission to access to database and execute transaction. Authentication of users and application check in this step. User input username and password and then algorithm compare with existing username and password and then allow entering the system and access to data, which has permission.

(ii) **Step2. Execute transaction:** Now user start executing transaction. The system and algorithm does not now about users transaction, because in this process users does not define their transactions.

(iii) **Step3. Logging file:** while user executing the transactions the system monitor all users or application behavior and log all activity in log file.

(iv) **Step4. Runtime analyzer:** This step run while a transaction is executed as parallel. After each transactions activity this step check transaction and analyze it. After each reading log file information save in audit table.

(v) **Step5. Black list check:** the transactions compare with a data storage which called black list.

This data storage is created by algorithm during executing detection process (process two) to have a faster search and prevent malicious transaction. Data that saved in black list are query, operation, tables, filed, and username.

(vi) **Step6: policy check:** after analyze transaction algorithm check and compare analyzed transaction with policy data storage that defines by database administrator.

(vii) **Step7. Graph Dependency:** check: like step six, algorithm compares simulated transaction with this data storage. Graph dependency show relationship between tables in database. in this table audit table compare with graph dependency table, if fields in transaction that are saved in audit table is equal with fields in graph dependency then check table name in graph dependency. If user has not permission to access this table, algorithm prevent execute transaction.

(viii) **Step8. malicious check:** in this step that run after these three last steps if any misuse behavior discovered the algorithm doing these three things respectively. First determinate the transaction, second algorithm roll back database to the last integrity mode and finally if this transaction or task does not exist in black list, algorithm add it to the black list. Else algorithm accept transaction and allow terminate it.

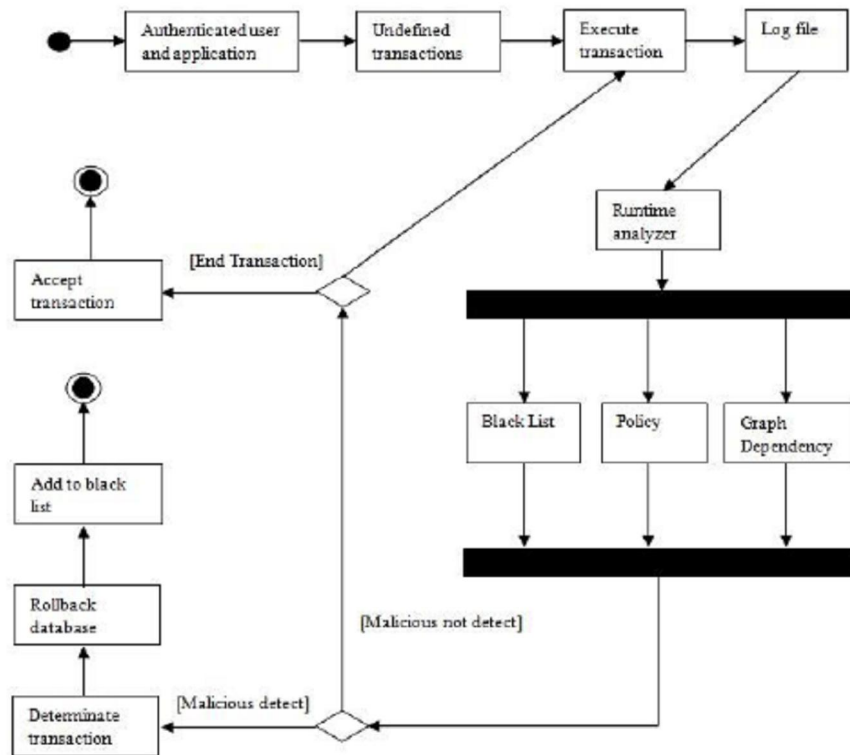


Figure 8: Detection process

## Results

The evaluation result indicated that the all effective detection and prevention is acceptable in detection and prevention of transaction malicious activity. This algorithm covers weakness of coverage percentage of detection malicious activity and false alarm. Also, this algorithm covers weakness of detection multi transaction activity to inference data that user has not allow to access.

Table 2: Comparison algorithm by false negative and coverage percentage

Algorithm		False negative percentage	Coverage percentage
Existing	Detection of Insiders Misuse in Database systems (DIM) (Shatnawiet <i>al.</i> , 2011)	43.	57
Proposed	Prevention and detection malicious Transaction (PDMT)	30	70

## Conclusion

Malicious transaction is a type of attack, which modifies data on database maliciously. A new type of this attack is done by multi transaction, which these transactions are normal one, but by running these transactions together to inference data that is not allow to access it. There are some method and algorithm to detect and prevent these malicious transactions but some of them have some limitations and some of them do not support new types of malicious. The proposed prevention and detection malicious transaction algorithm can detect and prevent as well the new type of attack that is multi transaction malicious by adding data storage (policy and black list). Also black list help algorithm to have a faster search to detect malicious transaction.

## References

- Chen, J., Lu, Y., & Xie, X. (2007). An Auto-Generating Approach of Transactions Profile Graph in Detection of Malicious Transactions. *Intelligent Information Hiding*, 1–4.
- Chinchani, R., Iyer, A., Ngo, H., and Upadhyaya, S. (2005). Towards A Theory Of Insider Threat Assessment. *International Conference on Dependable Systems and Networks (DSN'05)*.
- Chung, C. Y., Gertz, M., and Levitt, K. (2000). Demids: A misuse detection system for database systems. *Third International IFIP TC-11 WG11*. 5: 159-178.
- Heights, H. (2009). Insider Threat in Database Systems: Preventing Malicious Users ' Activities in. 1616-1620.
- Li, W., Panda, B., and Yaseen, Q. (2012). Malicious Users ' Transactions: Tackling Insider Threat. *Architecture*.
- Maybury, M., Chase, P., and Cheikes, B. (2005). Analysis and Detection of Malicious Insiders Sara Matzner1. The Threat: Malicious Insiders Figure 1. Heterogeneous and Multilevel Data Sources 4. Event and Observable Taxonomy. Decision Analysis.
- Mun, H., Han, K., Yeun, C. Y., and Kim, K. (2008). Yet Another Intrusion Detection System against Insider Attacks. *Proceesings, Symposium on Cryptography and Information Security*.
- Nguyen, N., Reiher, P., and Kuenning, G.H. (2003). Detecting Insider Threats by Monitoring System Call Activity. *Workshop on Information Assurance*.
- Nithiyandam, C., Tamilselvan, D., Balaji, S., and Sivaguru, V. (2012). Advanced framework of defense system for prevetion of insider's malicious behaviors. *2012 International Conference on Recent Trends in Information Technology*, 434–438. doi:10.1109/ICRTIT.2012.6206788
- Parveen, P., Evans, J., Thuraisingham, B., Hamlen, K. W., and Khan, L. (2011). Insider Threat Detection using Stream Mining and Graph Mining. *IEEE International Conference on Privacy, Security, Risk, and Trust*
- Rathod, Y.A., Chaudhari, M.B., and Jethava, G.B. (2012). Database Intrusion Detection by Transaction Signature. *ICCCNT'12 26th\_28th July 2012, Coimbatore, India, ieee*.
- Shatnawi, N., Althebyan, Q., and Mardini, W. (2011). Detection of Insiders Misuse in Database Systems. *Computer, I*.

8/21/2014