Feature Selection for Developing a Detection Strategy against Tamper of Grades from Malicious Insiders

John Roy A,. Geralde
Ateneo de Manila University
Quezon City, Philippines
+639228764488
jrageralde@addu.edu.ph

Kardi Teknomo, PhD
Ateneo de Manila University
Quezon City, Philippines
kteknomo@ateneo.edu

ABSTRACT

In this paper, we define the problem of detecting the security issue of data manipulation in a particular application of encoding of grades. The problem is related to the field of data mining of early warning systems. We enumerate the existing strategies and the solutions. As demonstration of our concept, an existing set of relational data was collected and analyzed to determine the factors to be used for detection.

Keywords:

Early Warning Systems, Data Mining, Relational Databases, Data Security and Integrity

1. INTRODUCTION

Data protection is a necessity among organizations using relational databases for information storage and retrieval. Malicious insider behavior is considered one of the gravest threats by institutions who have established database systems to support their regular operations. Malicious threats at data integrity can come from both external and internal sources. Between the two, the internal source is perceived to be more difficult to control and detect. Malicious intrusions are hard to detect because these things are done by authorized users within the scope of their assigned duties and responsibilities. In addition, certain types of attacks namely Tamperage and Fabrication are among the hardest to analyze and detect.

Tamperage is an update done on table rows to alter the true result. Examples of tampering are changing of student grades, altering client bank deposits, altering item balances in an inventory list, etc. Fabrication, on the other hand, is creation pieces of data that do not reflect a true event which entails insertion of invalid records. Fabrication can come in the form of creation of fake customer collections or fictitious orders to vendor, etc. Most incidents of tamperage and fabrication entail manipulation of data from tables that represent relationships in an E-R model.

Institutions adopt regular data audit systems and procedures in order to protect integrity of data. Audit procedures are regularly done in order to maintain integrity. However tamperage and fabrication are usually committed outside audit schedules for them to escape audit attention.

This paper intends to address these are the types of malicious control. We intend to select some features that would be used to develop detection strategy against tamper of student's grades from malicious insiders. To demonstrate our concept of feature selection, we choose data from Ateneo de Davao due its availability to the researchers.

As an educational institution, Ateneo de Davao University, keeps records of academic performance of students in electronic form. At present, the university uses relational database management software to store academic data. Like any other organization at present, it is subject to insiders threats of tamperage and fabrication. Actual cases have been observed and certain mechanisms are put in place to safeguard against it. However, at the present, no early warning mechanism has been devised to predict, analyze and avoid future abuse.

In this paper, we describe how to select the features that would help us to develop a detection criteria based on existing database logs from the university relational database in preparation for the development of an early warning system against particularly grade tamperage.

2. LITERATURE REVIEW

2.1 DeMIDS: A Misuse Detection System for Database Systems

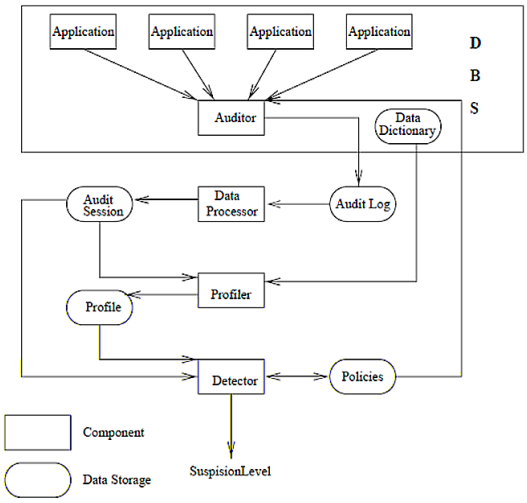

Figure 1: The DeMIDS Architecture (after [1])

The research approach uses a set of tools to derive user profiles from audit logs. Such profiles describe the typical behavior or access patterns of users in the system by specifying the typical values of features that are audited in the audit logs. The profiles derived are used to detect misuse behavior [1].

DeMIDS (DEtection of Malicious Insiders for Database Systems) consists of four components (see Figure 1), namely : (1) Auditor (2) Data Processor (3) Profiler and (4) Detector.

The Auditor is responsible for collecting the audit data of users by auditing their queries through the auditing functionality of the DBMS. A set of interesting features to audit is selected by the SSO, depending on the security policies to establish or verify. Monitored features are recorded in audit logs.

The Data Processor is responsible for preprocessing the raw data in the audit logs, such as handling missing values, converting the raw data into appropriate data structures and types for the Profiler.

The DeMIDS approach introduces the concept of working scopes of users which consists of attributes that are closely related in the database schema and are often referenced together in database retrieval and modification statements issued by a the user. The system then computes, for each working scope, the ratio between the shortest distance and the longest distance in the schema. Any issued that is not within the ratio of the working scope is considered outside of the scope and is a candidate for misuse.

2.2 Detecting Anomalous Patterns in Relational Databases

The research approach requires mining SQL queries stored in database audit logs. The result is used to form database access profiles that models normal database access behavior and identify intruders. In these research, two different scenarios are used namely databases using Role Based Access Control (RBAC) and databases whose users are not associated with roles. Clustering algorithms are used to group normal user behaviors. For detection, the clustered profiles are used as roles or employer outlier detection techniques to identify behaviors that deviate from profiles [2].

Unlike the approach used in DeMIDS this research based the profiles on groups. In addition, the research approach does not require knowledge of the underlying schema to compute distances among attributes in the users working space.

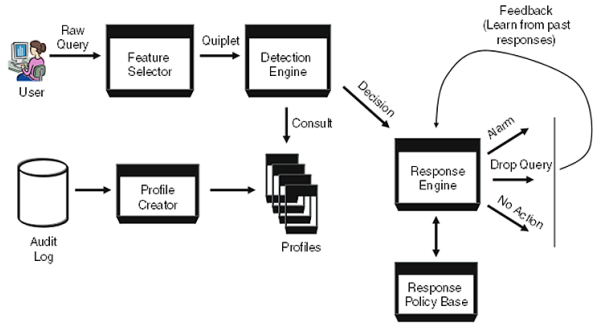

Figure 2: Overview of the Identification Process (after [2])

To build the profiles, the audit logs are preprocessed into a data structure that can be analyzed called a Quiplet. Each quiplet contains the following information: 1. The SQL command, 2. The Projection Relation Information, 3. Projection Attribute Information for each relation, 4. Selection Relation Information, 5. Selection Attribute Information.

Role Based anomaly detection proceeds using a data mining technique starting with Classification, experimental Evaluation and Anomalous Query Generation

2.3 PostGreSQL Anomalous Query Detector

The research of [3] aims to demonstrate the integration of DBMS specific Anomaly Detection (AD) mechanism within the core of the DBMS functionality. The research assumes that RBAC is supported by the underlying DBMS. As in the preceding research, the anomaly detection provides a profile for each role that represents accurate and consistent behavior of users holding the role. A data structure is established to represent the behaviors in a form that can be analyzed. The behavior is then evaluated against traces from the database audit logs that represent the true or normal behavior of the users. Likewise a classifier is trained using this behavior and used to identify anomalous behavior.

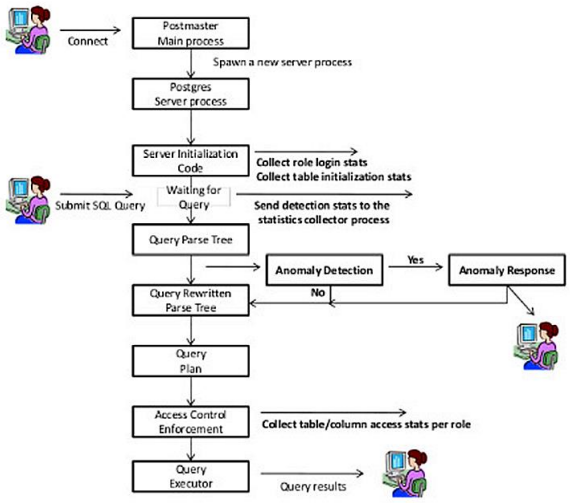

Figure 3: Anomaly Detection and Data Collection Hooks in PostgreSQL (after [3])

Assuming that users interact with the database using SQL commands, the research uses a triplet representation to capture the information of the input query. The data structure contains the SQL command, a binary vector called ACCREL-BIN which identifies the relations used by the SQL command, and another binary sub vector called PROJ-ATT-BIN which identifies the attributes used for each relation in the second component.

The research then proceeds with a data mining technique to segregate false positives and false negatives. The remaining components then are evaluated against the roles to identify candidates for anomalous behavior.

3. METHODOLOGY

3.1 Data Acquisition

Actual results of previous investigations involving actual tamperage are collected. The actual database grade edit logs are then collected and classified according to time edit distance, period of the actual editing and edit distance.

3.2 Features Selection

From the collected data logs, we derived several features that we suspect would help us to detect the malicious intent. We expect that the timing of student's grade updates to actual incidence of tamperage and fabrication would be the main features that will be used as basis for data analysis.

3.3 Frequency Distribution

The collected edit logs are the compared to actual investigation results to identify the true positives. Comparison of probability distributions between the tampered grades and the unmodified

grades based on the derived features is used to identify which features are significant in identifying the tamperage criteria.

4.0 Results & Discussion

4.1 The Acquired Data

An investigation was triggered by a teacher report on an erroneous grade and was carried out by selected university personnel to identify the truth and extent of the tamperage. The proponent sought permission to obtain the results and converted them into a database composing the positive results.

Edit Logs are obtained from the university database and compared with the investigation results. The tampered grades are then classified from the unmodified grades and tallied in a frequency distribution table.

4.2 Features Selection

Based on our prior knowledge, we expect that the following four derived features would be the main feasible features to detect tamperage. The main hypothesis lies on the determination of the timing of student's grade updates to actual incidence of tamperage and fabrication.

Using the data collected the following features were derived:

1. Days from Scheduled Encoding – The number of days of the date of editing and the actual date it is expected.
2. Day of Editing – the day that the update was done. The days are classified as follows: 1-Sunday, 2-Monday, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday and 7-Saturday
3. Period in the Day – The period in the day when the update as done and are classified as follows: Early Morning – before 9:00 am, Midday – from 9:00 am to 4:00 PM, Late Afternoon – from 4:00 pm onwards.
4. Period in the Semester – the period in the semester that the update was done. The period are classified as follows: 1 – No Classes, Period, 2. – Enrollment Period, 3- Prelims Period, 4 – Midterm Period, 5- Finals Period, and 6- Summer Period.

4.3 Frequency Distribution

4.3.1 Days from Schedule Encoding

Table 1 shows the distribution of tampered and unmodified grades based on the difference between number of days of the date of editing and the actual date it is expected. The results showed that a large number of the unmodified grades (6177 or 95%) were encoded less than 100 days from their expected date of encoding. On the other hand, a large number of the tampered grades (88 or 97%) were encoded 200 days beyond their expected date of encoding. We fail to reject our hypothesis that the following features are the time difference between the editing and the schedule of encoding is one of the feasible features to detect tamperage.

Table 1. Distribution of Editing Based on Time Differences of the schedule encoding

Time Difference (in Days)	Tampered Grades		Unmodified Grades	
	Count	%	Count	%
<100	3	3.30	6177	95.52
100-199	0	0	107	1.65
200-299	30	32.97	65	1.01
300-399	2	2.20	9	0.14
400-499	28	30.77	20	0.31
500-599	0	0	2	0.03
600-699	11	12.09	9	0.14
700-799	1	1.10	10	0.15
800-899	15	16.48	14	0.22
900-999	1	10.99	2	0.03
1000 above	0	0	52	0.03
	91		6167	

4.3.2 The Day of Editing

Table 2 shows that of the tampered grades, 56% were done on Friday and a large number (25%) were done on either a Tuesday. On the other hand, the distribution of the unmodified grades is more uniform during weekday and is actually lessen on Friday. The two distributions of tampered and unmodified grades are significantly difference and therefore we fail to reject our hypothesis that days in a week is also one of the feasible features to detect tamperage.

6.3.3 The Time of Editing

Table 3 showed that both modes of the distributions of tampered and unmodified grades are the same at mid-day. Thus, we reject our hypothesis that time of day is one of the features to detect tamperage.

Table 2. Distribution of Database Update Based on Day in a Week

Day	Tampered Grades		Unmodified Grades	
Sunday	Count	%	Count	%
Monday	9	9.89	1193	18.45

Tuesday	23	25.27	1443	22.31
Wednesday	7	7.69	1372	21.21
Thursday	1	1.10	1018	15.74
Friday	51	56.04	756	11.69
Saturday	0	0	685	10.59
	91		6467	

Table 3. Distribution of Editing Based on Time in a Day

TIME OF DAY	Tampered Grades		Unmodified Grades	
	Count	%	Count	%
EARLY AM	0	0	816	12.62
MIDDAY	62	68.13	5005	77.39
LATE PM	29	31.87	646	9.99
TOTAL	91		6467	

Table 4. Distribution of Editing Based on Period in Semester

PERIOD	Tampered Grades		Unmodified Grades	
	Count	%	Count	%
NO CLASSES	3	3.30	2414	37.33
ENROLLMENT	12	13.19	1210	18.71
PRELIM	68	74.72	314	4.85
MIDTERM	8	8.79	59	0.91
FINALS	0	0	2112	32.66
SUMMER	0	0	358	5.54
TOTAL	91		6467	

6.3.4 The Period in the Semester

Table 4 indicates that of the tampered grades a large number (74%) occurred during the Prelim period. A bigger number (87%) occurred during either Enrollment or Prelim period. On the other hand, less than 5% of the unmodified grades occurred during both the Prelim and Midterm Periods. A large percentage (95%) of the unmodified grades occurred when there are no classes or during Enrollment or Finals Period. Since the two distributions are significantly difference, we fail to reject our hypothesis that period in a semester is also one of the feasible features to detect tamperage.

5.0 CONCLUSIONS

The results of our investigation strengthen our hypothesis that the following features are the feasible features to detect tamperage:

1. The time difference between the date of editing and the legal date of editing. The longer the time difference, the higher the probability that tamperage may happen.
2. Day of editing. Higher probability happens on Tuesday and Friday.
3. Period in a semester. Enrollment or Prelim period has higher probability that tamperage may happen.

We also found out that time in a day does not contribute much in term of probability to detect tamperage.

The output of this research can be useful in devising an algorithm to detect malicious intent among database updates by insiders.

6.0 REFERENCES

[1] Christina Yip Chung, Michael Gertz, and Karl Levitt, Demids: A misuse detection system for database systems, In Third International IFIP TC-11 WG11.5 Working Conference on Integrity and Internal Control in Information Systems (1999), 159–178, Kluwer Academic Publishers.

[2] Ashish Kamra, Evimaria Terzi and Eliza Bertino, Detecting Anomalous Access Patterns in Relational Databases. (April, 2007) , Springer Verlag.

[3] Bilal Shebaro, Asmaa Sallamm, Ashish Kamra and Eliza Bertino, PostGre Anomalous Query Detector. (March 18-22, 2013) , Genoa, Italy, ACM 978-1-4503-1597-5/13/03.