www.ijecs.in

International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 2 Issue 4 April, 2013 Page No. 1162-1165

Protection of Data Base Security via Collaborative Inference Detection

Alok Kumar Shukla, Ajay Sharma, Dharmendra Kumar

(shukla.shuklaalok.alok@gmail.com, ajayitmit.01@gmail.com, dharmak21@gmail.com)

B.Tech Final Year Student , Department of Information Technology & Engineering Institute of Technology And Management AL-1 Sector-7 Gida Gorakhpur Uttar Pradesh(India)

Abstract *- In many applications like Defense department, Commercial departments and Marketing departments we need a strongly secured database. Database securities are needed in order to protect our identity and authentication process of users. We propose a novel security mechanism to overcome inference problems and risks for securing the database. Our approach is used for the violation inference detection for single users and multi users. An agent is located between the user input query and the database. Our approach can be used for both the single user as well as the multiple users. This process achieves high authorization, communication accuracy and trust in communication and preventing data from leakage by inference. Here Work is focused on employee information access. Probability of each employee goes on increasing on each query request. When a user poses a query, detection system will examine users past query log for last three days and calculates probability. If probability exceeds than the specified threshold, the query will be denied for that day. Also, to monitor activities, security officer can generate log.*

Index Terms: *knowledge processing; database; inference; probability; protection; security; query;*

I. Introduction

The restrictions for protecting a database system are generally stated in terms of database views, which can be used by a user to access a relation which he is not authorized to directly access. However, with information flow and inference, a user may acquire additional, unauthorized information, which may be the exact values of attribute or the relationship of attributes. Two inference techniques can be used to derive additional information: (1) analyzing functional dependencies between attributes within a relation or across relations, (2) merging views with the same constraints. On the other hand, information flows are caused by invoking a sequence of queries to indirectly read/write the sensitive data of a relation. Our approach is used to defend users from accessing the secured data from the data source or data centers. But the former techniques are not fulfilled because intruders can acquire the inoffensive information and apply inference techniques to execute sensitive information by the data. We construct a novel technique as inference detection system which is used to identify the inference problem and inference risks. Our approach monitors the trespassers from the database. When a new query is posted, we can compare the query with the query history. Query will be unresponsed if the probability of the query exceeds the threshold of the sensitive information. This actuates us to lead our inquiry from a singleuser case to a multiple-user case, where users may cooperate to each other to collectively deduce conscious data. We have carried a set of researches by using our inference trespass detector as a testbed to understand the characteristics in coactions and the effect on collaborative inference. Thus, coactions inference for a specific task can be derived by tracking the query history of all the users together with their collaboration levels (CLs).

Ii. Work Done

In analyzing different inference in the database system, researchers have expanded their an inference project on eliminating the issue with varies techniques and persuaders. In recent years, researchers have found a method that prevents inference within databases from recurring in the system. By locating inference channel and preventing any occurrence of these types of problems happening in the system. Some have used semantic data modeling to detect the inference channel. It looks into database design and redesigns it to make sure that this type of inference does not occur in the system. The other technique evaluates the database system, which read, update or both by using database transaction to determine if inference has occurred.

The technique will either disable the query or reclassify the query in higher level, only if it discovers an illegal inference. Several techniques have been addressed for the inference problems in the database Rule-based inference schemes were employed in this paradigm to defend the protection, since data modify can regard data inference, a performance that spreads modify to the user history files to assure that no question is refused based on the noncurrent entropy. To cut down the time in testing the entire account login calculation illation a anterior cognition of data addiction to cut down the search space of a coition and thus dilute the execution time for illation. The previous work on data inference mainly focused on deterministic inference channels such as functional dependencies. The knowledge is represented as rules, and the rule body exactly determines the rule head. Although such rules are able to derive sound and complete inference, much valuable nondeterministic correlation in data is ignored.

As shown in figure 2.1, when any user fires a query, inference detection module will check probability of that user from past log as well as data probability. If probability is below 0.6 or data probability is below 0.8 then only that user will have access to data, otherwise query access will be denied.

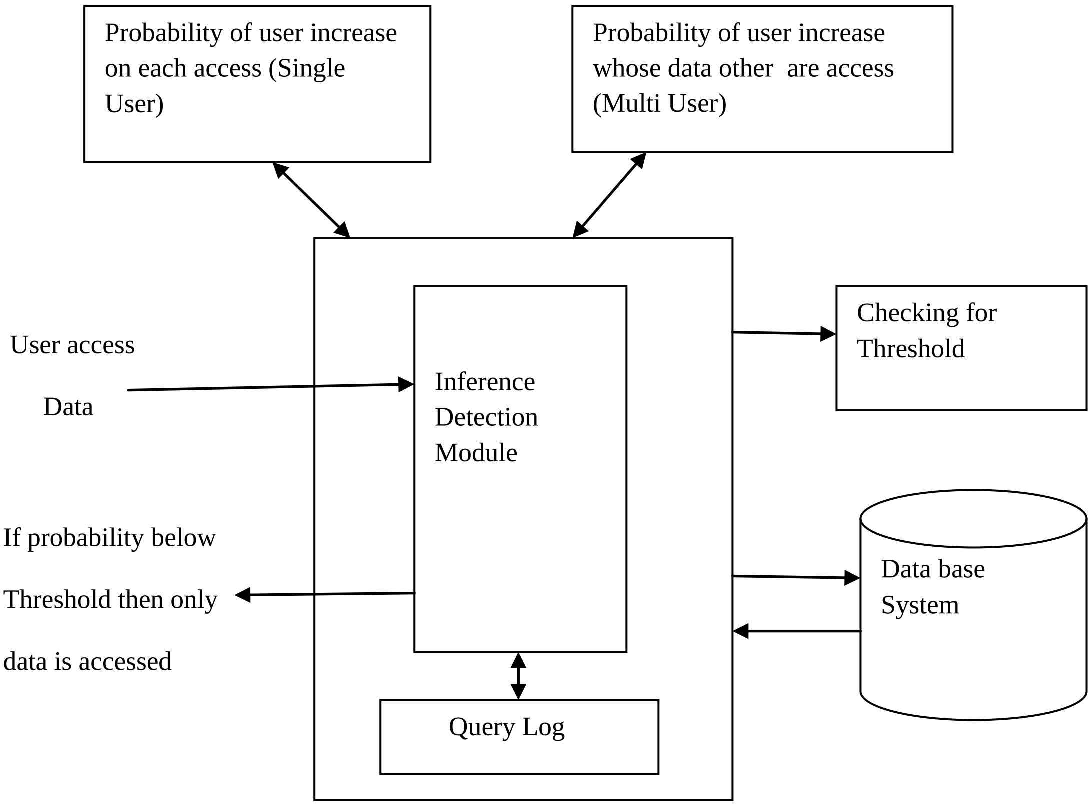
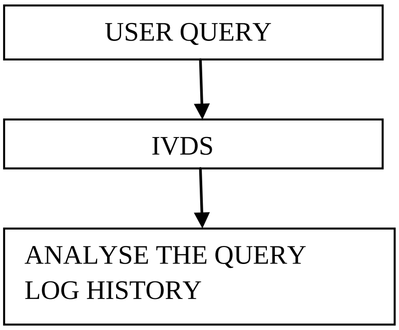

Probability of user increase on each access (Single User)
Probability of user increase whose data other are access (Multi User)
User access
Data
Inference Detection Module
Checking for Threshold
If probability below
Threshold then only
data is accessed
Data base System
Query Log

Fig .1 Inference Detection Module

Iii. Process Of Ivds

In our process we extend our research model from single user to multiple user to inference secured data. we develop IVDS to identify the cooperation between the users and the information flows based on the cooperation. The cooperative inference for a specified query is based on the query history for all the user with their cooperation levels. IVDS sets the threshold for every current queries, the IVDS gets the query result from the query log.



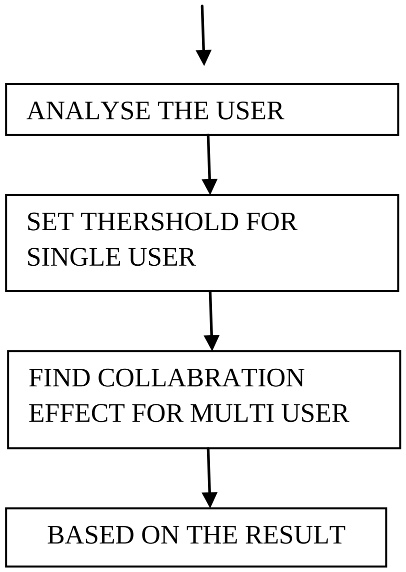

ANALYSE THE USER
SET THERSHOLD FOR SINGLE USER
FIND COLLABRATION EFFECT FOR MULTI USER
BASED ON THE RESULT

Fig 2. Overall process of IVDS

Further, many semantic relationships, as well as data mining rules, cannot be specified deterministically. To remedy this shortcoming, we propose a probabilistic inference approach to treat the querytime inference detection problem. The contribution of the paper consists of 1) deriving probabilistic data dependency, relational database schema, and domain-specific semantic knowledge and representing them as probabilistic inference channels in a SIM, 2) mapping the instantiated SIM into a Bayesian network for efficient and scalable inference computation, and 3) proposing an inference detection framework for multiple collaborative users. System owners have been required to put in place very rigid requirements for keeping their systems fully compliant with strict security policies and to have their systems scanned on a regular basis to guarantee that no security configuration has been altered. This strict security requirement has been accentuated by several government laws, regulations, directives, and publications.

Iv. Inference Infraction Discovery for Single User

IVDS provide an integrated view of the relationships among data attributes, which can be used to detect inference violation for sensitive nodes. In such a graph, the values of the attributes are set according to the answers of the previous posted queries. Based on the list of queries and the user who posted those queries, the value of the inference will be modified accordingly. If the current query answer can infer the sensitive information greater than the pre specified threshold, then the request for accessing the query answer will be denied. The notion of imbedding policies into the database itself and altering these policies to closure every try to determine the land of the database, or to vary its shape in a way that opposes what has been accomplished and fed into the policy by the system owner. These policies can be accomplished at different graininess levels in such a way that the system owner can choose to raise coarse-grained policies to supervise and control the behavior of the database as a whole through the use of global settings, or invoke fine-grained policies that affect specific aspects or configuration settings. But the absolute core principle of our framework is the notion that the security policies, as well as all the database objects and logic that enforces them, are made an integral and inseparable part of the database that they are meant to protect.

V. Inference Infraction Discovery for Multi User

Generalizing from the single-user collaborative system to the multiuser collaborative system greatly increases the complexity and presents two challenges for building the inference detection system. First, we need to estimate the effectiveness of collaboration among users, which involves such factors as the authoritativeness of the collaborators, the communication mode among collaborators, and the honesty of the collaboration. In addition, we need to properly integrate the knowledge from collaborators on the inference channels for the inference probability computation. Database administrators or power users can alter security configurations in a way that could result in unauthorized access to and compromise of the database. An example would be that of granting privileged access to unprivileged users, or just simply misusing his privileged access. Another example is one that pertains to security scans or audits of the database. Independent auditors are usually hired to perform a security scan of the database and they work with the DBA to get the database to a point where it is hardened enough to pass the scan. However, a database administrator can temporarily (or permanently) set some or all of the configuration parameters back to their original settings in order to achieve certain goals that he thinks are justified. The DBA can easily set that parameter to unlimited, change the password to the same one, and then set that parameter back to what it is supposed to be. By doing so, the DBA would have violated the rule that applies to reusing the same password over and over again. In this paper we describe a policy based approach for enforcing database configurations even to those who have privileged access. We do not advocate minimizing the role of the DBA or restricting his access. However, we do advocate that each action gets verified and approved by system owner embedded, predefined configuration policies before it is applied to the database. Unlike database security frameworks that exist today, which mostly detect imminent problems, generate an alert, and produce a report, our solution, which is an inseparable component of the database that it is meant to protect, mitigates any detected risk on its own without having to wait for human intervention.

Vi. Collabration Effectivness

We shall define CL as a metric for measuring the percentage of useful information flow from the information source to the recipient. The range of CL is from 0 to 1. CL = 0 and CL = 1 mean that none or all of the information is received by the recipient. By a series of experimental studies, we find that the CL depends on three components: the authoritativeness of the information provider A, the honesty of the collaboration H, and the fidelity of the communication channel between the provider and recipient F. The authoritativeness of the information provider represents how accurate the information is. If a provider is knowledgeable and has high reputation in the field related with the task, then he/she can provide more accurate information. Honesty represents the honesty level of the provider and his/her willingness of releasing his/her knowledge to the recipient. For example, if user A is very knowledgeable, and A and B have a good communication channel, then both the authoritativeness and fidelity of user A are high. However, A is not willing to release his full knowledge to B. As a result, the useful information cannot reach B for inference. Further, A can deceive B with false information. Thus, we shall use the honesty measure as an indication of the honesty in collaboration.

Fidelity measures the effectiveness of the communication between the provider and recipient. Poor mode of communication can cause information loss during the transmission, which reduces the effectiveness of the collaboration. Authoritativeness measures how accurate the provider can supply information, honesty describes the willingness of the provider to release the accurate information, and fidelity measures the percentage of information transferred to the recipient due to the limitation of the communication mode. Once we estimate these three components for a set of users on a specific task.

Vii. Conclusion And Future Research

In this paper we have implemented a technique to protect sensitive information content. Malicious users can exploit the correlation among data to infer sensitive information from a series of seemingly innocuous data accesses. This developed inference detection system can be used for any organization with very small changes as per their database. Its ability to detect inference at the early stage rather than detecting after the attack is already committed. The developed Semantic Inference Model works for single user as well as for multi user environment. The developed system can be successfully deployed in any industry to deal with the threats that pose from internal users in an attempt to secure sensitive information. Further research and experiment in use of nested queries and use of multiple relations is needed. The inference problem is a very harmful effect in securing the database. The attack may be happened along with the database architecture and the major consequences are handled by the database maintaining servers. For this we designed the IVDS (Inference Violation Detection System) which evaluates the query posted by every user and based on the analysis history of the every query (backlog) we can specify whether the IVDS answers the query or deny the query. This approach can be applied for both the single user as well as the multi users. We evaluate our approach in the real time experiments and obtain the results by giving various queries and different levels of users.

References

1. C. Farkas and S. Jajodia,(2002), “ The Inference Problem: A Survey,” SIGKDD Explorations, vol. 4, no. 2, pp. 6-11.

2. Alexander Brodsky, Csilla Farkas, and Sushil Jajodia, (2000), “Secure Databases: Constraints, Inference Channels, and Monitoring Disclosures”.

3. Delugauch, Harry S., and Thomas H. Hinke. “Wizard: A Database Inference Analysis and Detection System.”

4. Chavira, M., Allen, D., Darwiche, A.: Exploiting Evidence in Probabilistic Inference.

5. Y. Chen and W. W. Chu. “Database Security Protection via Inference Detection.”