Journal of Applied Science and Engineering Innovation, Vol.2 No.11 2015, pp. 451-454
ISSN (Print): 2331-9062
ISSN (Online): 2331-9070

Research on Multilevel Secure Database Inference Control

Junsheng Ma[1], Mingshun Xing[1], Xiaoshuang Wang[1], Jin Meng[1], Haiyan Zhao[1]

[1] *Xi'an Communication Institute, Xi'an 710106, China*

Abstract*:* In multi-level security database system, the inference problem is that a low security level user using accessible data and his own knowledge and to infer confidential information of higher security level, which constitute an attack on the database. This paper describes the common inference channel. By adding inference control module in the database design and operation stages to detect and control inference, our research aimed at improving the security of the database.

Keywords data security inference channels multi-level security database system

INTRODUCTION

Multi-level security database protects sensitive information and data integrity mainly through mandatory access control mechanisms, which ensures that host of low-level safety level, can not directly obtain unauthorized information, but it does not ensure the information stored in the database must be secured. By combining information obtained within his security level and external knowledge, low security level host can infer unauthorized information. Therefore, it constitutes an attack on the database, which is named the multi-level secure database inference problem. Using inference method to obtain unauthorized information is a more covert method to attack. Database systems with high level of security requirements should consider defense against such attack.

FORMAL DESCRIPTION OF INFERENCE PROBLEM

Research on formalizing the inference problem is focus on the definition of the inference problem. The researches are mainly conducted from these dimensions: set theory, classical information theory, data partitioning and functional dependencies, etc.

Description of set theory

Consider in a database, each data item is assigned an access level, and assume that the level of access is a collection of partial order. Define the relationship of the following:

Given a data item x and y, x → y indicates that y can be inferred from x.

Relation → is reflexive and transitive.

S is a closed set of inference. If x belongs to the set S, and x → y holds, then y also belongs to S.

For an access level L, let E (L) represents a collection of responses to all the possible access whose level L is less than or equal to L. If E (L) is not inferring closed, then there is an inference channel.

A classic description of information theory

Given two data items x and y, and let H (y) represents the uncertainty of y and let Hx (y) indicates the uncertainty of y given the condition of x (The uncertainty here is to define by the conventional information theory). Therefore, the decrement of the uncertainty of y, after x is given, can then be defined as follows:

$\mathrm{INFER}\ (x \rightarrow y) = (H\ (y) - Hx\ (y))\ /\ H\ (y)$

In the formula:

INFER (x → y) - Decrement of the uncertainty of y after x is given

H (y) - Uncertainty of y

Hx (y) - The uncertainty of y after s is given.

The value of INFER (x → y) is between 0-1. If the value is 0, there is no information about y can be inferred from x. If the value is between 0 and 1, it is possible to infer information of y given x. If the value is 1, x can certainly infer y.

Description of data partitioning

For each user, the data in the database can be divided into two partitions: a visible and an invisible set of data. User is only allowed to access the visible data. The access to the invisible data set is not allowed. 'Known' represents the amount of data items that the user is already known. The set is from previous query result. If the intersection of set 'Invisible' and set 'Known' is empty, there is no inference problem. Otherwise, there is inference problem.

Description of functional dependencies

Functional dependency is defined as follows:

Let R be a relational schema. * denotes its set of attributes. X, Y ⊆ *. When any two tuples u, v whose corresponding values of those properties in both components of X is equal, there is u, v whose corresponding Y component of those attributes is equal. i.e., if u [X] = v [X], then u [Y] = v [Y]. It is called function X determines y, or function y depends on X.

COMMON INFERENCE CHANNELS AND SOLUTION

Inference using logical relationship between multiple results of queries.

Firstly, the attacker issues multiple database queries, such as averages, total, etc., and then analyze the query results to infer information of high security level.

In statistical databases, users are generally allowed to query the type of gathering information (such as total, average, etc.), but not allowed to query a single record information. For example, query "average salary of cadres" is allowed, but the query "salary of cadres A" is not allowed.

In order to describe the inference channel clearly, here are some examples:

User A would like to know the salary of user B's. A can get the result through the next two legal queries:

(1) Total salary of A and other N cadres.

(2) Total salary of B and the same other N cadres.

Assuming the result of the first query is X, result of the second query is Y, as user A knows his salary is Z, then he can calculate the salary of user B= Y-(X-Z).

The key point of this inference channels is that there are many duplicated data (i.e., the other N-salary) between the two queries. Therefore, we can restrict the intersection between any two query can not exceed M. This makes it more difficult to get the data of other users. It can proved that [3] in the provision above, if A want to known the salary of user B, user A needs to make at least 1 + (N-2) / M queries.

We can continue to restrict that the number of queries for each user can not exceed 1 + (N-2) / M, but if there are two users collaborating, this restriction is still ineffective.

Inference through graded constraints

Graded constraints are rules describing data classification standards. If these grading standards are informed, the user may infer from these constraints to obtain sensitive data.

For example:

Suppose a company is divided into three levels of cadres, company level, vice company level and platoon level. Salary of same level is same. If user A wants to know the salary of cadres B. A can make the following query:

User A: Is B a company level cadre?

System: Access denied.

User A: Is B a vice company level cadre?

System: It is not.

User A: Is B a platoon level cadre?

System: It is not.

User A noticed that the system is trying to hide the level of user B. If B is not a company level cadre, the system would have answered "no" to the first question, instead of "Access denied." So B is certainly company level cadre.

Sicherman, de Jonge and van de Riet considered the protection of confidential information when the grading constrain is known or unknown by the attacker. They defined lots of conditions to determining whether to deny access. They described a set of strategies based on these conditions. And then they developed a formal model of secured database with safety rules.

Inference through functional dependencies between different levels of data

There are commonly "functional dependency" and "multi-valued dependency" relationships between data tables. These dependencies may produce inference channel.

Suppose in a military database, three items are stored about cadres: name, rank and salary. Name and level are not considered sensitive data, while salary is sensitive data. The same level cadres have the same salary. So in this case, the access of sensitive data is denied. The user can easily know the salary of other cadres, and yet salaries are sensitive data. Therefore an inference channel exists.

In the example above, the reason why the inference channel exists is when determining the security level of levels and salary, the functional dependencies between levels and salary: level → salary is not considered. If a user knows the level of a cadre, the salary of this cadre is also known by this user. In this case, the solution is to upgrade the security level of cadre level, from the non-confidential level to confidential level. If the attribute can be dependent on the function in a manner consistent with the security tag, these problems can be solved.

Inference though value constraints

A value constrain is constrain of data with one or more data items. If a constraint is defined on the data with different levels of security, it can be used to produce an inference channel [5].

For example, suppose that A are not confidential data and B is confidential data. If the database constrains $A + B \leq 100$ mandatorily, which is set to be not confidential, and any user can have access. The value of B does not directly affect the value A, but it can affect the set of possible values of A's. Thus an inference channel is created. If a constraint is defined on several security levels, it must be separated into several single security level constraints. For this example, the constraint $A + B \leq 100$ should be divided into $A \leq 50$ and $B \leq 50$, avoiding the inference channel.

INFERENCE CONTROL IN DATABASE DESIGN STAGE

Inference control module (Figure 1) is added in the database design. User can only access the data after examined by inference control module. Inference control module determines whether the user can infer sensitive information based on the results already obtained by the user. If it is possible to infer sensitive information, there may be inference channel. The system will deny current access, and modify database objects security classification mark, to control the database inference channel. If the modification is unsuccessful, the functional dependency will be sent to the database run-time inference control module as a database runtime inference control rule.

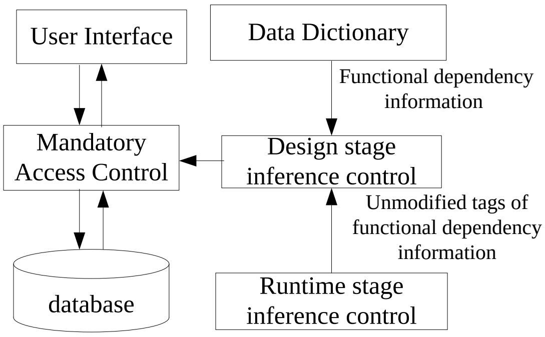
User Interface
Data Dictionary
Functional dependency information
Mandatory Access Control
Design stage inference control
Unmodified tags of functional dependency information
database
Runtime stage inference control

Figure 1 database system with design stage inference control module

Multi-level secure database design stage inference control mainly consists of the following modules:

(1) Data Dictionary modules: Data Dictionary module stores large amounts of sensitive information, as well as possible sensitive information rules, functional dependencies, value constraint relationships and grading constraint relations. Processing a query, inference control module first perform a inference using the user's current an previous query results based on knowledge in the data dictionary module to determine if the user can obtain sensitive information. If it is possible to infer sensitive information, deny access.

(2) Mandatory Access Interface modules: inference control module obtains safety rules and system security design information in database security module through mandatory access interfaces. Meanwhile the security classification mark of main object is also obtained by forced access interface module. If possible inference channels found, mandatory access interface module can also modify the corresponding security classification mark. For the relationship to be examined, information to obtain is : a set of functional dependencies in the security tag and keywords for each attribute defined on the set of relations.

(3) Inference Control Module: inference control module analyze current and previous query results, to determine whether it is possible for sensitive information to be inferred. If possible, there may be an inference channel, therefore the system would deny the access, and prompts the system security administrator to modify the security classification mark to control the database inference channel in the database. If the modification is unsuccessful, the functional dependency will be sent to the database run-time inference control module as a database runtime inference control rule.

(4) Run-time inference control module interface: Inference control module in design stage can not fully detect and eliminate inference channel, which requires inference control modules in runtime as complement, eliminating inference channels. When an inference control module in design stage found that there may be an inference channel, but it can not modify its security tag, the system would record these functional dependencies, adding to the library of inference control module rules through interface on runtime inference control module, for the run-time inference control.

DATABASE RUNTIME STAGE

By adding the inference in the database runtime control module (Figure 2), inference control module analyzes the queries and query results submitted by users, detecting inference channels according to the inference rules stored in the system, performing inference control according to the results of analysis. For the inference problems can only obtain the external knowledge in database runtime stage, they can be controlled only in database runtime stage. Meanwhile, in runtime stage, the system also needs to detect and control the inference channels that were not eliminated during database design stage.

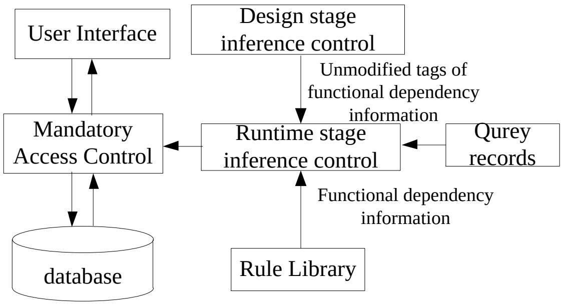
User Interface
Design stage inference control
Unmodified tags of functional dependency information
Mandatory Access Control
Runtime stage inference control
Qurey records
Functional dependency information
database
Rule Library

Figure 2 database systems with inference control module in runtime

Multilevel secure database runtime inference control mainly consists of the following module:

(1) Design Inference Control Interface Module: For those inference channel that were not fully controlled in design stage, and unsuccessfully modified security marks, their functional dependencies must be sent through the design stage

inference control module to the run-time stage inference control module which performs modification and controlling.

(2) Mandatory Access Interface Module: Similar to mandatory access interface module in design stage

(3) Inference control module: inference control module uses the inference rules for multiple user queries for analysis, to see whether the user can obtain sensitive information, and if so, to prove the existence of a potential inference channel, the system will reject the user's query, the user returns the empty set.

(4) Query history module: detection due to the operation of inference repeatedly query results should be compared in order to determine whether there is inference channel, so saving a record period of time is needed.

(5) Rule library modules: run-time inference control designers need to use a lot of inference rules provided for the user to analyze the results of multiple queries to determine if there inference channel, so the run-time inference control modules need to add a rule base, with to hold the inference rules.

CONCLUSIONS

Inference channel is a very important, but relatively speaking, not fully resolved problem in database security. It is impossible to obtain a fully general solution to the inference problem. Some scholars proposed some effective solutions to common inference problems, constructing models to detect and eliminate some inference channel. However, due to the diversity of inference method and universally of information available for inference, inference problem becomes very complex. Existing technologies can only detect and eliminate some certain inference. Current technologies and solutions are mostly theoretical models. There is rarely a practical tool. Meanwhile, the presence of data mining techniques makes the database inference problem has become a more serious problem than ever before. Data mining has changed the situation that any attacker must rely on there personal experience and knowledge to analyze the data. Powerful data mining tool allows the user to process the results from a database in a very efficient, simple, intelligent and automatic way, finding valuable information. Therefore, the inference problems caused by data mining will become the new focus and hotspot of research in this area.

by running get access interface module, user inquiries via mandatory access control checks by the run-time inference control module for further examination.

REFERENCES

A Evfimievsk,i J Gehrke, R Srikant. Limiting privacy breaches in privacy preserving data mining [J] . In Proceedings of the 22nd Symposium on Principles of Database Systems, ACM Press, 2003: 211- 222

J Domingo-Ferrer. Advances in inference control in statistical databases: Anoverview in inference control in statistical databases: from theory to practice [J]. LNCS2316, Springer-Verlag, 2002:1-7

Jajodia S, Meadows C. Inference problems in multilevel secure database management systems[A]. Abrams M, Jajodia S, Podell H, eds. Information Security: An Integrated Collection of Essays[C]. Los Alamitos: IEEE Computer Society Press, 1995，570-584

L Wang, D Wijesekera, J Sush i.l Cardina lity-based inference control in sum - only data cubes[J]. In Proceedings of the 7th European Symposium on Research in Computer Security, 2002

Li Lixin, Liao Changrong, Chen Weiming, etal. Inference attack analysis of MLS DBMS using rough set theory[A], 1n: Proc. Of 7th Joint Ititl. Computer Conf. Shantou, China, 2001. 1339-1452