Foundations for an Access Control Model for Privacy Preservation in Multi-Relational Association Rule Mining

Stanley R. M. Oliveira[1,2]
oliveira@cs.ualberta.ca

Osmar R. Zaïane[2]
zaiane@cs.ualberta.ca

[1]Embrapa Information Technology
André Tosello, 209 – Barão Geraldo
13083-886 - Campinas, SP, Brasil

[2]Department of Computing Science
University of Alberta
Edmonton, AB, Canada, T6G 2E8

Abstract

Recent data mining algorithms have been designed for application domains that involve several types of objects stored in multiple relations in relational databases. This fact has motivated the increasing number of successful applications of relational data mining over recent years. On the other hand, such applications have introduced a new threat to privacy and information security since from non-sensitive data one is able to infer sensitive information, including personal information, facts or even patterns that are not supposed to be disclosed. The existing access control models adopted to successfully manage the access of information in complex systems present some limitations in the context of data mining tasks. The main reason is that such models were designed to protect the access to explicit data (e.g. tables, attributes, views, etc), whereas data mining tasks deal with the discovery of implicit data (e.g. patterns). In this paper, we take a first step toward an access control model for ensuring privacy in relational data mining, notably in multi-relational association rules (MRAR). In this model, users associated with different mining access levels, even using the same algorithm, are allowed to mine different sets of association rules. We provide the groundwork to build our access control model over existing technologies and discuss some directions for future work.

Keywords: Access Control; Mining Access Control; Privacy Preserving Data Mining, Privacy Preservation in Association Rule Mining; Security; Data Mining.

1 Introduction

Most data mining techniques have been developed for extracting patterns and trends in the traditional matrix form, in which the rows correspond to observations and the columns represent variables. This representation has been the traditional one in statistics and has some advantages, such as the succinct data analytic procedures and the possibility to devise efficient algorithms (Džeroski & Lavrač 2001). However, data in the real world is seldom of this form. Rather, applications in the real world contain various types of entities involved in multiple tables in relational databases. Thus, the discovery of hidden patterns and trends can be done directly in multiple relations without manual preprocessing to transform the data into a single table.

Relational representation has some advantages over single-table representation. For instance, in database terminology, a single-table representation is said to be a non-normal form database and is considered bad database practice (Ramakrishnan & Gehrke 2000). In (Wrobel 2001), Wrobel discusses some problems inherent in the single-table representation (e.g. redundancy of information, problems with update) that can be addressed by relational databases, which are able to represent information as a set of different interlinked tables.

The practice of relational representation has influenced the number of successful applications of relational data mining over the recent years (Srikant & Agrawal 1996, Fu & Han 1995, Džeroski & Lavrač 2001). The successful results obtained have led researchers within the information security community to investigate the impact of data mining technology on database security (Clifton & Marks 1996, Johnsten & Raghavan 1999, Chang & Moskowitz 2000). Such an investigation considers how much information can be inferred or calculated from large data repositories made available through data mining algorithms and looks for ways to minimize the leakage of information, a topic which has also been investigated in the statistical databases area (Willenborg & Waal 1996, Castano, Fugini, Martella & Samarati 1995).

Despite its benefits in various areas such as marketing, business, medical analysis, bioinformatics and others, data mining can also pose a threat to privacy in database security if not done or used properly. Recent advances in data mining and machine learning algorithms have introduced new problems in database security (Johnsten & Raghavan 1999, Clifton 2000). The main problem is that from non-sensitive information or unclassified data, one is able to infer sensitive information, including personal information, facts, or even patterns that are not supposed to be disclosed.

In the context of relational databases, one of the most known approach to protect information relies on Role-Based Access Control (RBAC) (Sandhu, Coyne, Feinstein & Youman 1996) which prevents users from obtaining a sufficiently large and varied sample of a database. Although RBAC ensures that only authorized users are given access to certain data or resources, it does not prevent users from finding patterns that they are not supposed to discover using data mining. The reason is simple: RBAC was designed to enforce security on explicit data, and data mining techniques' goal is to find hidden patterns that rely on implicit data. Because data mining introduces new problems in relational databases, the problem of inference has received considerable attention in the database security community. Preventing this type of inference detection is beyond the reach of the existing methods. Therefore, the integration between database security and data mining remains a fertile area for future research.

In this paper, we provide a basic theory that per-

mits one to develop an access control model for multi-relational association rules (MRAR), which can be built over existing database technologies. One major novelty with our approach is that we take into account the concept of mining level, i.e., users are associated with different mining levels and, even using the same association rule algorithm, they are able to mine different set of association rules.

We benefit from the work on database security (Castano, Fugini, Martella & Samarati 1995). Of particular interest is work on multilevel relations in the context of multilevel secure databases (Jajodia, Samarati, Sapino & Subrahmanian 2001). It allows multiple levels of security to be mapped into multiple mining levels. In doing so, a user is allowed to mine association rules only in his or her mining level. The idea behind multiple levels also received a special attention in the recent International Conference on Very Large Data Bases (28th VLDB). Particularly, the authors in (Agrawal, Kiernan, Srikant & Xu 2002) claim that the future directions in security database research include features, such as searches on encrypted data, access control based on multiple levels containing a special attribute, called "purpose", which is similar to attaching security level with records in secure databases.

Our access control model can be integrated with a Decision Support System (DSS) or any systems designed to support decision making in the context of relational data mining. In such systems, there is frequent access of information for mining purposes but not as many updates. To date, such schemata have not been explored in detail.

The effort described in this paper is by no means meant to be complete or comprehensive. Rather, our primary goal is to present our preliminary ideas in order to motivate discussion on richer access control models for MRAR. We argue that this line of work will eventually lead to a formal model. However, first we must develop the conceptual foundations for such a model.

This paper is organized as follows. In Section 2, we provide the basic concepts of existing access control models and MRAR as well as the definition of the research problem. In Section 3, we describe the necessary security requirements for an access control model for MRAR. In Section 4, we introduce our access control model for MRAR. Related work is reviewed in Section 5. Finally, Section 6 presents our conclusions and a discussion of future work.

2 Basic Concepts

2.1 Access Control Models

Several advances have been made in the area of access control specification. Policies have been devised to narrow the gap between what the security administrators want and what the access control system can provide. The most common approaches are Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role-Based Access Control (RBAC) (Castano, Fugini, Martella & Samarati 1995, Sandhu, Coyne, Feinstein & Youman 1996, Ferraiolo & Kuhn 1995).

DAC models control the access to the information explicitly specifying the authorization for each user to each resource in the system. The access control is at the discretion of the object's owner or anyone else who is authorized to control the information object's access. Rights can be passed from one subject (also called user) to another. The obvious advantage of DAC is that it is extremely flexible. However, DAC does not provide real assurance on the flow of information in a system. It is possible to bypass the access restrictions stated through authorizations. For instance, a user who is permitted to read data can pass it to others who are not authorized to read it without the acknowledgment of the data's owner. Considering that the dissemination of information is not controlled in DAC, this makes this approach vulnerable to malicious attacks such as Trojan Horses embedded programs (Castano, Fugini, Martella & Samarati 1995). Thus, DAC seems to be proper to environments in which information sharing is more important than protection of information.

MAC models are aimed at addressing the leakage of information which is present in DAC models. In MAC, access control decisions are made beyond the control of the individual owner of the object. A central authority (security administrator) determines what information is to be accessible by whom, and the users cannot change the rights. The advantages of MAC models derive basically from their suitability to some kinds of environment in which the users and objects can be assigned to a security clearance or a security level. In this approach, a user can only read a resource of lower security clearance and the user can only write to a resource with a higher security clearance. Although the write restriction may seem counterintuitive at first, this restriction is necessary to guarantee that information only flows upwards in security clearance.

The RBAC framework is based on the set of entities: users, roles and access permissions (also called authorizations). A user can be also represented by a group, or even a program executing on behalf of a user. A user can be a member of one or more roles. The notion of role is an enterprise or organizational concept, i.e., a role represents a job function in an organization and embodies a specific set of authorizations and responsibilities for the job. Similarly, a role can have many permissions and the same permissions can be assigned to many roles. Thus, a role can inherit permissions assigned to another role in a role hierarchy since role hierarchies are natural means for structuring roles to reflect an organization's lines of authority. Permissions are the rules that describe how the objects (e.g. tables, attributes, views) are accessed by users.

System administrators can create roles, grant permissions to those roles, and then assign users to roles on the basis of their specific job responsibilities (Ferraiolo & Kuhn 1995). This greatly simplifies the management of access rights. For this reason, RBAC has been very attractive for several kinds of applications, such as commercial, governmental, corporate intranet, among others. The main reason is that RBAC models are capable of reducing complexity and cost of security administration.

2.2 Basics of MRAR

One of the most studied problems in data mining is the process of discovering association rules. The discovery of interesting association rules among huge amounts of data can be very effective in revealing actionable knowledge that leads to strategic decisions.

In multi-relational data mining, the data model consists of several relations (tables) in which each of them describes particular objects' features, but only one view of the objects is central to the analysis. The important point here is that this only view is obtained dynamically without data preparation to squeeze as much relevant data as possible into a single table.

One very efficient alternative for mining association rules in multiple relations is to use meta-rules (Han & Kamber 2001). Meta-rule-guided mining of association rules allows users to specify the syntactic form of rules that they are interested in mining. Formally, a meta-rule is a rule template in the form of $P_1 \wedge P_2 \wedge ... \wedge P_m \rightarrow Q_1 \wedge Q_2 \wedge ... \wedge Q_n$, where P_i (for $i = 1, ..., m$) and Q_j (for $j = 1, ..., n$) are either instantiated predicates or predicate variables. A rule R_C complies with a meta-rule R_M, if and only if it can be unified with R_M.

Suppose that a portion of the relational schema of a store of material for hiking trips is presented as follows:

customers(cno, name, rating, age, occupation, city)
items(ino, item_name, price)
buys(cno, ino, date, qty, total)

Meta-rules can be used to find multidimensional association rules in large relational databases. Multidimensional association rules can be categorized into three groups as follows: (1) *Single-Dimensional* association rules (also called Intradimensional) contain a single distinct predicate with multiple occurrences, such as $buys(c, \text{“}Ski\ pants'') \rightarrow buys(c, \text{“}Sunglasses'')$; (2) *Multidimensional* (also called Interdimensional) which contain no repeated predicates, such as $rating(c, \text{“}Excellent'') \wedge age(c, \text{“}20..23'') \rightarrow buys(c, \text{“}Gloves'')$; and (3) *Hybrid-Dimensional* association rules contain repetition of some predicates, such as $buys(c, \text{“}Gloves'') \wedge occupation(c, \text{“}Student'') \rightarrow buys(c, \text{“}Sunglasses'')$.

The attributes present in these relations can be classified in categorical or quantitative. Categorical attributes have a finite number of possible values with no ordering among the values (e.g. rating, occupation). Categorical attributes are also called nominal attributes since their values are names of things. On the other hand, quantitative attributes are numeric and have an implicit ordering among values (e.g. age, price).

Note that Single-Dimensional association rules are mined from single tables, while Multi-Dimensional and Hybrid-Dimensional association rules may involve join(s) of more than one relation.

2.3 Privacy Preservation Problem in MRAR

The specific problem addressed in this paper can be stated as follows: If D is a relational database or even a data warehouse and M is the set of all association rules that could be mined from D, the goal is to provide users of different levels of access to D so that for each level i, the corresponding users are able to mine a set of association rules M_i, such that $M_i \subseteq M$.

For instance, let us consider a company in which there are three levels of access to information for mining relational association rules. Considering the hierarchy in which $Level_1 > Level_2 > Level_3$, the set of association rules that could be mined from these levels must hold $M_1 > M_2 > M_3$, and M_1 is the set all association rules that could be mined from a relational database D.

Let us consider the privacy problem described in (Du & Atallah 2001). In this problem, two or more companies plan to cooperatively work on a project for their mutual benefit. Thus, each organization would like its own requirements to be satisfied. However, their requirements are proprietary data which include the customer's projects of the likely future evolution of certain commodity prices, interest and inflation rates, economic statistics, portfolio holdings. Therefore, nobody likes to disclose its requirements to the other party, or even to a trusted third party. How could they cooperate on this project while preserving the privacy of the individual information?

For this problem, such companies may have only two levels of access to information. The top level for the business purpose of each company, and the second level for data exchange with the company's partner.

3 Requirements of an Access Control Model for MRAR

In this section, we analyze the necessary requirements for an access control model for MRAR in enterprise environment. We start by discussing the general requirements for access control models, followed by an analysis of security requirements for an access control model for MRAR.

3.1 General Requirements for Access Control Models

Type of Policy: laws according to which accesses are controlled. Policy in access control models can be classified into mandatory (MAC models), discretionary (DAC models), role-based (RBAC) (Castano, Fugini, Martella & Samarati 1995).

Target System: some MAC models are designed for operating systems protection such as Bell-LaPadula (Bell & La Padula 1973) and Biba (Biba 1977) models , while others are designed for database security such as Sea View model (Denning et al. 1988). In general, DAC models were designed for both operating systems protection and database security. RBAC focus mainly on database security.

Type of Control: Some models are oriented to direct access control such as the DAC models Access Matrix (Graham & Denning 1972), Take-Grant (Jones 1978), and Action-Entity (Bussolati, Fugini & Martella 1983), while MAC models allow both direct access control and indirect access or information-flow control. RBAC models are designed for direct access control only.

Addressed Security Aspects: the most important information security aspects in databases include secrecy and integrity (Ramakrishnan & Gehrke 2000). Ensuring secrecy means preventing, detecting or deterring the improper disclosure of information, whereas ensuring integrity means preventing, detecting or deterring the improper modification of information. Considering that secrecy and integrity are related to database security, these security aspects are not present in DAC models. However, they are present in most MAC model as well as in RBAC models.

The summary of the general requirements for access control models can be seen in Table 1, including the requirements for a MRAR models discussed in Section 3.2.1.

3.2 Security Requirements for an Access Control for MRAR

In this section, first we analyze how the general requirements in Section 3.1 fit to the access control model for MRAR, and then we discuss additional security requirements.

Table 1: Summary of the General Requirements for Access Control Models

Requirement	DAC	MAC	RBAC	MRAR Model
Type of Policy	Discretionary	Mandatory	Role-Based	Mandatory
Target System	Part OS & Part DB	OS & DB	DB	DB
Type of Control	Access	Access & Flow	Access	Access & Flow
Security Aspects	None	Secrecy & Integrity	Secrecy & Integrity	Secrecy & Integrity

3.2.1 General Requirements for an Access Control for MRAR

Type of Policy: Mandatory policy since it contains security labels (e.g. mining dimensions) for objects and users.

Target System: Privacy preservation in MRAR, in the context of relational databases.

Type of Control: Oriented to control both flow of information and access control without update operations (e.g. write, insert, delete).

Security Aspects: The focus is mainly on secrecy, however integrity is also guaranteed implicitly since this access control model does not allow users to update the data.

3.2.2 Additional Requirements for an Access Control for MRAR

Req1: The access control for MRAR must be based on a hierarchy of security levels, in which each security level corresponds to a mining level.

Req2: Users associated with a certain mining level cannot pass rights to users assigned with other mining levels. Changing this security requirement induces an outflow of information and violates some axioms presented in Section 4.4.

Req3: If one user is authorized to access one mining level and this mining level contains another mining level, then the user is also allowed to access the contained mining level. This assumes a hierarchy of mining levels. We call this property *subsumption of rights*.

Req4: Users are granted rights only to access parts of the data they need to perform their mining tasks. This is similar to the principle of the least privilege in RBAC (Sandhu, Coyne, Feinstein & Youman 1996).

Req5: The access control for MRAR might be able to deal with multiple users mining concurrently, even though users perform mining tasks sporadically.

Req6: The capacity of a mining level cannot be exceeded by an additional mining level member. This is similar to the principle of the cardinality in RBAC (Ferraiolo & Kuhn 1995).

Req7: A user can never have an active mining level that is not authorized for that user.

Req8: A user can perform an operation (e.g. reading, mining) only if the operation is authorized for the mining level in which the user is currently active.

4 Top-MRAR Model

In this section, we introduce our access control model for MRAR, denoted Top-MRAR, which is designed to have hierarchical mining access levels that meet privacy preservation requirements.

4.1 Identifying Users and Mining Levels

Knowing who the users are and how they use the data is key to design the mining levels. One good strategy for identifying the proper number of mining levels is to design the levels in a process oriented view (e.g. affinity analysis, classification and regression rules, etc). By doing so, users can be associated with the levels according to their responsibilities.

Affinity analysis is distinct from association rules for prediction in terms of the language of expression and application as well, i.e., an affinity rule has the form: *When Item1 Also Item2*. An example of this is, *When "Ski pants" Also "Gloves"*. These rules can be mined from single tables (e.g. our previous "buys" relation). Affinity analysis is very useful in market basket analysis.

Classification and regression rules are widely used for prediction. For instance, one rule could be: "If *age* is between 35 and 50 and *occupation* is professor, then *rating* is excellent." This rule can be modeled as a multidimensional association rule, such as $age(c, \text{“}35..50''') \wedge occupation(c, \text{“}Professor'') \rightarrow rating(c, \text{“}Excellent'')$.

Now, suppose we design one mining level allowing its users to mine affinity rules from singles tables. Apparently, RBAC addresses this situation since users are granted rights to access parts of the data they need to perform their tasks. Typically, this is an all-or-nothing security approach which is easy to implement, but not astutely useful. However, in the context of MRAR the leakage of information becomes real when someone wants to prevent a group of users to mine a set of restrictive affinity rules from their view. Preventing this type of leakage of information is beyond the reach of the existing access control methods. Even limiting the access to the data, users may mine some of these restrictive rules from their view. In this case, there is a real need for security mechanisms which are able to hide such restrictive rules and, most importantly, this situation leads one to devise a distinct mining level for an access control for MRAR.

Note that the number of mining levels should be flexible since one company may require two or three mining levels, while others may require a more fine mining level granularity.

We illustrate this idea taking into account a situation in which three mining levels are required as follows: (1) *Full Mining* (FM) which allows users mining all kind of association rules (e.g. affinity rules, sequential patterns, classification and regression rules, etc); (2) *Specific Mining* (SM) in which users are allowed to mine affinity association rules; and (3) *Restrictive Mining* (RM) in which only a subset of affinity association rules can be mined from SM level, i.e., the set of restrictive rules are not available for this restrictive mining level. In this case, the following relation must hold: $Level_{FM} > Level_{SM} > Level_{RM}$. We provide one example of such levels in Section 4.2.

4.2 Basic Definitions of Top-MRAR

The Top-MRAR model is based on three sets of entities: *users*, *mining levels*, and *permissions*. Each

mining level is assigned to at least one *permission* and each *user* is associated with only one *mining level*.

Definition 1 *The Top-MRAR model is defined as follows:*

- *U, O, P, and ML (users, objects, permissions, and mining level respectively).*
- *permission:* $O \times U \times ML \rightarrow \{yes, no\}$*, a function that answers if a user is given some permission for mining a particular object at a given mining level.*

A *user* represents an individual, a group, or a program on behalf of a user. Each *user* has a unique identifier for authentication purpose. An *object* represents the passive entities of the system (e.g. tables, attributes, views, tuples). *Permissions* (also called authorizations) are the rules that describe how the objects are accessed by users. The value of access is either *yes* or *no*. An *operation* is the action on which the permission is defined, such as *reading*, *mining*, etc.

Considering that Top-MRAR model is designed to enforce privacy in MRAR, this model deals with data that has multilevel access. Thus, our model extends the concept of relation to include mining levels. Such levels can be assigned to attributes and tuples of a relation. We define a multilevel mining relation as follows.

Definition 2 *Let* $R(A_1 : D_1, [ML_1], ..., A_n : D_n, [ML_n], T_{ML})$ *be a multilevel relation schema, and for each* A_i*,* $1 \leq i \leq n$*, let* D_i *be the set of values associated with the domain named* D_i*,* ML_i *the mining level label for the attribute* A_i*, and* T_{ML} *the mining access level for the whole tuple. An instance of R that satisfies the domain in the schema is a set of tuples with n fields:*
$\{\langle A_1 : d_1, [ml_1], ..., A_n : d_n, [ml_n], t_{ML}\rangle \mid \forall i\ d_i \in D_i,\ ml_i \in ML_i;\ and\ t_{ML} \in T_{ML}\}$.

Reusing our example in Section 4.1, we considered a situation in which three different mining levels may be required: *Full Mining* (FM), *Specific Mining* (SM), and *Restrictive Mining* (RM). For this example, both ML_i and T_{ML} take the values {FM, SM, RM}, and $\forall_i\ T_{ML}$ must dominate ML_i. The notation $[ML_i]$ means that the mining level label may exist or not. To illustrate the concepts defined above, let us consider one instance of the relational schema presented in Section 2.2. The corresponding multilevel relation is given in Table 2.

For this example, only the attributes *CNO*, *INO* and *Total* have the mining level label. Table 2 contains the same view for both FM and SM mining levels. On the other hand, the view corresponding to RM level is depicted in Table 3. Note that for this level, the tuples whose TID are 500 and 700 are not displayed since T_{ML} must dominate any ML_i.

Another important observation is that the mining level labels (ML_i) control the access to the attributes that must be displayed, while the mining access level of tuples (T_{ML}) selects the tuples that have to be displayed.

Based on Table 3, we should point out that Top-MRAR model removes the whole tuples instead of releasing some attributes with values and others with missing values. The reason is that releasing missing values may violate some security requirements since by using classification or regression algorithms one may predict the missing values.

In our previous work (Oliveira & Zaïane 2002), we have developed a taxonomy of sanitizing algorithms that can be applied to SM level's views to generated RM level's views. Our sanitizing algorithms can be easily modified to the context of MRAR. Removing a item from a pattern is equivalent to removing a tuple of a multilevel relation. Our algorithms deal with this situation properly and the example in Table 3 reflects this idea.

4.3 The Top-MRAR Structure

The Top-MRAR model is composed of three layers as can be seen in Figure 1. The *Authenticator* is the first layer that requires proof of identity. This is achieved by using mechanisms such as userid/password. The second layer, *Checker*, will store the access control mechanisms and the permissions are designed based on identity. After authenticating a user, the *Checker* layer applies some incoming queries before passing the queries to the database. The users will only access to objects (e.g. tables, attributes, views) that are compatible with their rights. In the last layer, the server is an existing database server.

The assumptions for this model are the following: (a) All connections to the database server have to pass through the *Checker* layer; (b) Existing access control system in the database server may continue to be in place; and (c) The relational database integrated to the *Checker* layer does not deal with frequent updates, except for *append* operation.

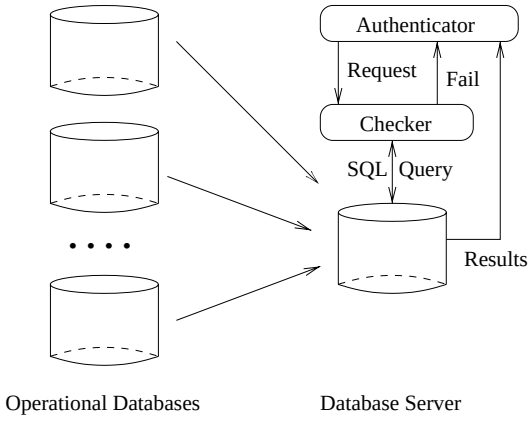

Figure 1: The Top-MRAR Structure

As illustrated in Figure 1, the operational databases may represent the branches of a business organization and the database server, integrated to the Top-MRAR model, might be used for mining purposes.

Note that, in this model, all queries will pass through the *Checker* layer. Based on the permission of a user, the *Checker* layer enforces that the results sent by the database server to the front-end will guarantee the security specifications. To accomplish that, the *Checker* layer works as follows: (1) queries that fail the security specifications are dropped and the *Checker* layer will return one message of access violation to the front-end; (2) users' queries that meet the security specifications of the system are directly passed to database back-end.

We should point out that the *Checker* layer does not discard access control strategies that may exist in the database server. Rather, the SQL queries still need to pass through the existing access control mechanism of the database. For instance, the database server may use RBAC. So the *Checker* layer may pass the RBAC checks before returning the results of the queries.

Table 2: An Example of a Multilevel Relation

TID	CNO	ML_{CNO}	INO	ML_{INO}	Date	Qty	Total	ML_{Total}	T_{ML}
100	C1	RM	I2	RM	01/05/2001	1	165.00	SM	RM
200	C1	RM	I4	RM	01/05/2001	2	60.00	SM	RM
300	C3	RM	I1	RM	01/06/2001	1	80.00	SM	RM
400	C3	RM	I3	SM	01/06/2001	1	120.00	SM	RM
500	C3	RM	I5	SM	01/06/2001	3	75.00	SM	SM
600	C4	RM	I3	SM	01/07/2001	1	120.00	SM	RM
700	C4	RM	I5	SM	01/07/2001	2	50.00	SM	SM

Table 3: The multilevel relation corresponding to users of the RM level

TID	CNO	ML_{CNO}	INO	ML_{INO}	Date	Qty	T_{ML}
100	C1	RM	I2	RM	01/05/2001	1	RM
200	C1	RM	I4	RM	01/05/2001	2	RM
300	C3	RM	I1	RM	01/06/2001	1	RM
400	C3	RM	I3	RM	01/06/2001	1	RM
600	C4	RM	I3	RM	01/07/2001	1	RM

4.4 Basic Properties

Execution of Top-MRAR model is governed by a set of axioms which must be satisfied in order for the system to be secure. Such axioms regulate the access of users to information in the database server for mining purposes. We say that the system is secure if, and only if, it satisfies these axioms. The following axioms are designed for the Top-MRAR model:

Mandatory Property: access of users to objects is governed by security labels (mining levels) on the users and objects. Usually centrally controlled by a security administrator function. No user is allowed either to modify his mining level or objects' mining level.

Membership Property: a user is supposed to be a member of only one mining level.

Append Property: append information is permitted without seeing its content, i.e., writing without reading. The append operation must be carried out by a security administrator in a specific time.

Read Property: a query from a user at a given mining level can access information from the database whose label is dominated by that level. The *Read Access* property corresponds to the No Read-Up principle in Bell-LaPadula model (Bell & La Padula 1973).

Mining Property: this property is completely related to *Read Access*, i.e., the information that can be read must be available for mining.

Non-Update Property: users are not allowed to alter data (e.g. insert, delete, and update) regardless of their mining level. The only operations permitted to users are *read* and *mining*.

Reclassification Property: In case of reclassification, a user at a given mining level must move to a upper level. This property prevents the system from indirect communication channels and keeps the information-flow control consistent.

Polyinstantiation Property: polyinstantiation occurs when there are multiples instances of data at different mining access level. This property is necessary to hide the existence of a higher level data from low users. Revealing the existence of a higher mining dimension object creates a covert channel.

5 Related Work

Some effort has been made to investigate the impact of data mining technology on database security (Clifton & Marks 1996, Johnsten & Raghavan 1999, Chang & Moskowitz 2000). Such investigations consider how much information can be inferred or calculated from large data repositories made available through data mining algorithms and looks for ways to minimize the leakage of information. This effort has been restricted basically to classification and association rules. In this work, we focus on the latter category.

Atallah et al. (Atallah, Bertino, Elmagarmid, Ibrahim & Verykios 1999) considered the problem of limiting disclosure of sensitive rules, aiming at selectively hiding some frequent itemsets from large databases with as little impact on other, non-sensitive frequent itemsets as possible. Specifically, the authors dealt with the problem of modifying a given database so that the support of a given set of sensitive rules, mined from the database, decreases below the minimum support value. This work was extended in (Dasseni, Verykios, Elmagarmid & Bertino 2001), in which Dasseni et al. investigated confidentiality issues of a broad category of association rules. This solution requires CPU-intensive algorithms and, in some way, modifies true data values and relationships.

In the same direction, Saygin et al. (Saygin, Verykios & Clifton 2001) introduced a method for selectively removing individual values from a database to prevent the discovery of a set of rules, while preserving the data for other applications. They proposed some algorithms to obscure a given set of sensitive rules by replacing known values with unknowns, while minimizing the side effects on non-sensitive rules.

Oliveira and Zaïane (Oliveira & Zaïane 2002) introduced a unified framework that combines techniques for efficiently hiding restrictive patterns: a transaction retrieval engine relying on an inverted file and Boolean queries; and a set of algorithms to "sanitize" a database. Specifically, this framework hides restrictive patterns without adding noise to the original data when sanitizing a transactional database, and considers the impact in the original database by quantifying how much information is preserved after sanitizing a database.

Related to privacy preserving data mining, but in another direction, Evfimievski et al. (Evfimievski, Srikant, Agrawal & Gehrke 2002) proposed a framework for mining association rules from transactions consisting of categorical items in which the data has

been randomized to preserve privacy of individual transactions. Although this strategy is feasible to recover association rules and preserve privacy using a straightforward uniform randomization, it introduces some false drops and may lead a miner to find associations rules that are not supposed to be discovered.

Rizvi and Haritsa (Rizvi & Haritsa 2002) proposed a scheme, based on probabilistic distortion of used data, composed of a privacy metric and an analytical formula. Although this framework provides a high degree of privacy to the user and retain a high level of accuracy in the mining results, mining the distorted database can be, apart from being error-prone, significantly more expensive in terms of both time and space as compared to mining the original database.

In the context of distributed data mining, Kantarcioglu and Clifton (Kantarcioglu & Clifton 2002) addressed secure mining of association rules over horizontally partitioned data. This approach considers the discovery of associations in transactions that are split across sites, without revealing the contents of individual transactions. This method is based on secure multi-party computation (Du & Atallah 2001) and incorporates cryptographic techniques to minimize the information shared, while adding little overhead to the mining task.

In (Vaidya & Clifton 2002), Vaidya and Clifton addressed the problem of association rule mining in which transactions are distributed across sources. In this approach, each site holds some attributes of each transaction, and the sites wish to collaborate to identify globally valid associations rules. This technique is also based on secure multi-party computation.

Recent directions in database security have pointed out the need for combining solutions to address complex issues, such as privacy preservation, the inference problem in databases, among others (Agrawal et al. 2002). Solutions to address these problems include techniques from statistical database (e.g. suppression, data swapping, etc) (Willenborg & Waal 1996, Castano, Fugini, Martella & Samarati 1995); access control models (e.g. multilevel relations) (Castano, Fugini, Martella & Samarati 1995); and the integration of cryptography and information retrieval (e.g. searches on encrypted data). In particular, the idea behind access control based on multiple levels relies on a special attribute, called "purpose", which is similar to attaching security level with records in secure databases. This idea is similar to that one in mining level, supported by our framework.

Our work differs from the related work in some aspects, as follows: First, we address the problem of privacy preserving in MRAR. To our best knowledge, this problem has not been considered in the literature so far. Second, our framework efficiently combines security features of existing access control models, and most importantly, it can be built over rather than replacing the existing access control models. Another important difference of our framework from the related work is that our focus is not only on privacy preserving on MRAR but also on maximizing the discovery of association rules in all mining levels, while minimizing the leakage of information. In addition, our framework does not require transformation of data into a single table.

6 Conclusions

In this paper, we have established the groundwork to build an access control model for multi-relational association rules (MRAR) over existing database technologies, called Top-MRAR model. Although the work described in this paper is preliminary and conceptual in nature, it is a vital prerequisite for the eventual development of a formal model.

This design of Top-MRAR greatly minimizes complexity for DBMS implementers since this model is built over rather than replacing SQL facilities. The proposed facilities can provide significant benefits for administering permissions in relational data mining, more specifically in the context of MRAR. Apart from these benefits, the implementation of our model seems feasible since our model inherits some features from existing access control models, such as the notion of multilevel relations, hierarchical security levels as well as some basic security requirements.

We reuse some security mechanisms from the work on database security, particularly the idea behind multilevel relations in the context of multilevel secure databases. Typically, we map security levels into multiple mining levels. In doing so, a user is allowed to mine association rules only in her mining level.

Our model is composed of three layers: the *Authenticator* which requires proof of identity, *Checker* that stores the access control mechanisms and permissions based on identity, and the *database server* that relies on an existing relational database. This access control model can be integrated with a decision support system (DSS) or any systems in which there is frequent access of information for mining purposes but not as many updates. In the context of our model, users are provided with views of the data and not the association rules, so that they are free to use their own association mining algorithms since the restriction for privacy in MRAR is applied before the mining phase.

The main contributions of this paper are as follows: (a) we analyzed the necessary security requirements for an access control model for MRAR; (b) we designed the framework structure of the Top-MRAR considering the integration with existing technologies; and (c) we provided the conceptual foundations and introduced basic definitions of our model.

Currently, we are studying new features that may be added to Top-MRAR model. More precisely, we are formalizing our model and extending it to encompass other data mining tasks such as classification, regression and clustering. Further work is needed to determine adequate ways of handling these mining tasks. We are also analyzing a way to integrate the mining levels with roles without violating the information-flow access, which requires further exploration.

7 Acknowledgments

Stanley Oliveira was partially supported by CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) of Ministry for Science and Technology of Brazil, under Grant No. 200077/00-7. Osmar Zaïane was partially supported by a Research Grant from NSERC, Canada. The authors would like to acknowledge the helpful comments made by the anonymous reviewers of this paper.

References

Agrawal, R., Kiernan, J., Srikant, R. & Xu, Y. (2002), Hippocratic Databases, *in* '28th International Conference on Very Large Data Bases', Hong Kong, China.

Atallah, M., Bertino, E., Elmagarmid, A. K., Ibrahim, M. & Verykios, V. S. (1999), Disclosure

Limitation of Sensitive Rules, *in* 'IEEE Knowledge and Data Engineering Workshop', Chicago, Illinois, USA, pp. 45–52.

Bell, D. E. & La Padula, L. J. (1973), Secure Computer Systems: Mathematical Foundations. ESD-TR-73-278, vol. 1-2, ESD/AFSC, Hanscom AFB, (MTR-2547, vol. 1-2, The MITRE Corp., Bedford, MA).

Biba, K. J. (1977), Integrity Considerations for Secure Computer Systems. ESD-TR-76-372, ESD/AFSC, Hanscom AFB, (MTR-3153, The MITRE Corp., Bedford, MA).

Bussolati, U., Fugini, M. G. & Martella, G. (1983), A Conceptual Framework for Security Systems: The Action-Entity Model, *in* '9th IFIP World Conference', Paris, France, pp. 127–132.

Castano, S., Fugini, M., Martella, G. & Samarati, P. (1995), *Database Security*, Addison-Wesley Longman Limited, England.

Chang, L. & Moskowitz, I. S. (2000), An Integrated Framework for Database Privacy Protection, *in* '14h Annual IFIP WG 11.3 Working Conference on Database Security', Schoorl, The Netherlands, pp. 161–172.

Clifton, C. & Marks, D. (1996), Security and Privacy Implications of Data Mining, *in* 'Workshop on Data Mining and Knowledge Discovery', Montreal, Canada, pp. 15–19.

Clifton, C. (2000), 'Using Sample Size to Limit Exposure to Data Mining', *Journal of Computer Security* **8**(4), 281–307.

Dasseni, E., Verykios, V. S., Elmagarmid, A. K. & Bertino, E. (2001), Hiding Association Rules by Using Confidence and Support, *in* '4th Information Hiding Workshop', Pittsburg, PA, USA, pp. 369–383.

Denning et al. (1988), The Sea View Security Model, *in* 'IEEE Symposium on Security and Privacy', Oakland, CA, USA, pp. 218–233.

Du, W. & Atallah, M. J. (2001), Secure Multi-Party Computation Problems and their Applications: A Review and Open Problems, *in* '10th ACM/SIGSAC New Security Paradigms Workshop', Cloudcroft, New Mexico, pp. 13–22.

Džeroski, S. & Lavrač, N. (2001), *Relational Data Mining*, Springer-Verlag, Germany.

Evfimievski, A., Srikant, R., Agrawal, R. & Gehrke, J. (2002), Privacy Preserving Mining of Association Rules, *in* '8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', Edmonton, AB, Canada, pp. 217–228.

Ferraiolo, D. F. & Kuhn, R. (1995), Role-Based Access Control: Features and Motivations, *in* '11th Annual Computer Security Applications Conference', New Orleans, LA, USA, pp. 241–248.

Fu, Y. & Han, J. (1995), Meta-Rule-Guided Mining of Association Rules in Relational Databases, *in* 'International Workshop on Knowledge Discovery and Deductive and Object-Oriented Databases', Singapore, pp. 39–46.

Graham, G. S. & Denning, P. J. (1972), Protection - Principles and Practice, *in* 'AFIPS Spring Joint Computer Conference', Montvale, NJ, USA, volume 40, pp. 417–429.

Han, J. & Kamber, M. (2001), *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco, CA, USA.

Jajodia, S., Samarati, P., Sapino, M. L. & Subrahmanian, V. S. (2001), 'Flexible Support for Multiple Access Control Policies', *ACM Transactions on Database Systems* **26**(2), 214–260.

Johnsten, T. & Raghavan, V. V. (1999), Impact of Decision-Region Based Classification Mining Algorithms on Database Security, *in* '13th Annual IFIP WG 11.3 Working Conference on Database Security', Seattle, USA, pp. 177–191.

Jones, A. K. (1978), Protection Mechanism Models: Their Usefulness, *in* 'Foundations of Secure Computing', Academic Press, New York City, NY, USA, pp. 237–254.

Kantarcioglu, M. & Clifton, C. (2002), Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data, *in* 'ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery', Madison, Wisconsin, USA.

Oliveira, S. R. M. & Zaïane, O. R. (2002), Privacy Preserving Frequent Itemset Mining, Workshop on Privacy, Security and Data Mining, IEEE ICDM, Japan, December 2002, also available as technical report: A Framework for Enforcing Privacy in Mining Frequent Patterns, TR02-13, Department of Computing Science, University of Alberta, Canada.

Ramakrishnan, R. & Gehrke, J. (2000), *Database Management Systems*, second edition, McGraw-Hill.

Rizvi, S. J. & Haritsa, J. R. (2002), Maintaining Data Privacy in Association Rule Mining, *in* '28th International Conference on Very Large Data Bases', Hong Kong, China.

Sandhu, R. S., Coyne, E. J., Feinstein, H. L. & Youman, C. E. (1996), 'Role-Based Access Control Models', *IEEE Computer* **20**(2), 38–47.

Saygin, Y., Verykios, V. S. & Clifton, C. (2001), 'Using Unknowns to Prevent Discovery of Association Rules', *SIGMOD Record* **30**(4), 45–54.

Srikant, R. & Agrawal, R. (1996), Mining Quantitative Association Rules in Large Relational Tables, *in* 'ACM SIGMOD International Conference on Management of Data', Montreal, Canada, pp. 1–12.

Vaidya, J. & Clifton, C. (2002), Privacy Preserving Association Rules Mining in Vertically Partitioned Data, *in* '8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', Edmonton, AB, Canada, pp. 639–644.

Willenborg, L. & Waal, T. D. (1996), *Statistical Disclosure Control in Practice*, Springer-Verlag.

Wrobel, S. (2001), Inductive Logic Programming for Knowledge Discovery in Databases, Chapter 4 of "Relational Data Mining", S. Džeroski and N. Lavrač (eds.), Springer-Verlag, Germany, pp. 74–101.