Limiting Disclosure of Sensitive Attribute Values: An Active Approach

Junping Sun, Member IAENG *

Abstract—**As computer and other technologies advance and various data and information are pervasively available, preservation of private information embedded in volume data are facing more and more challenges than ever before. Although there are available mechanisms in database management systems limiting access to some sensitive and private information, some inference techniques might still be able to bypass the access control mechanisms to acquire these unauthorized information inappropriately. This paper will propose an active approach to limiting disclosure of sensitive and private information.**

Keywords: Preservation of Private and Sensitive Information, Limiting Disclosure of Sensitive Data

1 Introduction

As both the size and the number of databases are growing exponentially, it is not only important to manage access control of sensitive and private data [4] [5], but also critical to protect the privacy from data inference by obscure database queries [2] [6] [7]. Next the generalized disclosure scenery with respect to inference rules will be given, and its related problem statement will be defined correspondingly. Section 2 will discuss various types of inference rules leading to possible disclosure of sensitive and private information. Section 3 will give the algorithms that detect these rules as well as set of private attributes whose values might be disclosed. Section 4 will present an active model based trigger mechanism that will take appropriate actions when a query might access to some private attribute values. Section 5 is the conclusion.

*Nova Southeastern University, Graduate School of Computer and Information Sciences, Fort Lauderdale, Florida USA 33314 Tel/Fax: 954-262-2082/3915 Email: jps@nsu.nova.edu

1.1 Disclosure Scenery

Given a universal database schema $\mathbf{S}(A_1, A_2, ..., A_n)$ with a set of attributes $A_i \in \mathbb{A} = \mathbb{A}_{PU} \cup \mathbb{A}_{PR}$ where $\mathbb{A}_{PU}$ is a set of publicly accessible attributes, and $\mathbb{A}_{PR}$ set of private attributes accessible to an authorized group of users, assuming that there is a set of publicly known rules such as $R_i \in \mathbb{R}$:

$$P_i \longrightarrow A_j$$

where $P_i \in \mathbb{P}_{PU}$, $1 \leq i \leq n$ and $A_j \in \mathbb{A}_{PR}$, $1 \leq j \leq m$, if $\exists P_i \in \mathbb{P}_{PU}$ can be satisfied by the query predicates $P_k \in \mathbb{P}_{PU}$, $1 \leq i = k \leq n$ from a given a query $Q \in \mathbb{Q}$, then $\exists A_j \in \mathbb{A}_{PR}$ $V(A_j)$ (the value of attribute $A_j \in \mathbb{A}_{PR}$) will be disclosed.

Although a given database query Q is only allowed to access $A_i \in \mathbb{A}_{PU}$, $1 \leq i \neq j \leq n$, the values $V(A_j)$ of private attributes might be inferred from some publicly known rules by any unauthorized individuals,

$$R_i \in \mathbb{R} : P_i \longrightarrow A_j,$$

that is,

$$Q = \{A_i \mid P_k\} \cup \mathbb{R} \models A_j$$

where $A_i \in \mathbb{A}_{PU}$, $P_k \in \mathbb{P}_{PU}$, $A_j \in \mathbb{A}_{PR}$, and $1 \leq i \neq j \leq n, 1 \leq j \leq m$.

In order to address the issues from the inference queries, the following model is proposed.

1.2 Problem Statement

Given a database query $Q \in \mathbb{Q}$, an active model is proposed to analyze Q and to discover if there exist potential inferences on sensitive data in database $r(S)$.

The active model is a 4-tuple and defined as:

$\mathcal{T} = \{\mathcal{E}, \mathcal{C}, \mathcal{KB}, \mathcal{AC}\}$ where

ISBN: 978-988-17012-2-0

- $\mathcal{E} = \{\mathbb{Q}, \mathbb{I}, \mathbb{D}, \mathbb{U}\}$ where $\mathbb{Q}$ is a set of queries, $\mathbb{D}$ set of deletions, $\mathbb{I}$ set of insertions, and $\mathbb{U}$ set of updates. Without loss of generality, all of $\mathbb{D}, \mathbb{I}, \mathbb{U}$ are treated as $\mathbb{Q}$ in this paper.
- $\mathcal{C}$ is a set of conditions that triggers corresponding actions, which will be discussed in the later part of this paper.
- $\mathcal{KB} = \{\mathbb{R}, \mathbb{IC}\}$ is the knowledge base including a set of inference rules $\mathbb{R}$ and set of integrity constraints $\mathbb{IC}$.
- $\mathcal{AC}$ is a set of actions and defined as:
  $\mathcal{E} \times \mathcal{C} \times \mathcal{KB} \longmapsto \mathcal{AC}$
  where $\mathcal{AC} = \{\mathbb{AC}_b, \mathbb{AC}_p\}$, $\mathbb{AC}_b$ is the set of prior actions before an event $Q \in \mathbb{Q}$ can proceed, and $\mathbb{AC}_p$ set of post actions after $Q \in \mathbb{Q}$ is processed.

In order to take appropriate actions on the inference queries, the following will discuss various types of inference rules possible to be used to access sensitive and private attribute values.

2 Inference Rules

2.1 Functional Dependency as Inference Rule

Definition 2.1 *Let $r(S)$ be a relation on the schema $S(A_1, ..., A_n)$, $\exists i$ $A_i \in X \subseteq S$, $A_i \in Y \subseteq S$, $1 \leq i \leq n$, $r(S)$ satisfies the functional dependency $X \longrightarrow Y$, denoted $r(S) \vdash X \longrightarrow Y$, if*

$$\forall t, t' \in r(S)\ t[X] = t'[X] \Rightarrow t[Y] = t'[Y],$$

$$i.e., \forall x \in D(X), \quad \mid \pi_Y(\sigma_{X=x}(r(S)) \mid \ \leq\ 1$$

where $D(X)$ is the domain of attribute X, π is the project operator, and σ the select operation in relational model.

The functional dependency $X \longrightarrow Y$ claims that for each unique value of X, there exists at most one value of Y.

Definition 2.2 *Let $r(S)_H$ be a relation in a historical database H accessible to public.*

For given $r(S)$ and $r(S)_H$, there exists two tuples such as $t \in r(S)$ and $t' \in r(S)_H$. For a given database query Q, assuming Q is allowed to access only the set of attributes $A_i \in \mathbb{A}_{PU}$, $1 \leq i \leq n$, if there exists a functional dependency $X \longrightarrow Y$ where $X \subseteq \cup A_i \subseteq \mathbb{A}_{PU}$ and $Y \subseteq \mathbb{A}_{PR}$, then the value of Y can be inferred and will be disclosed because if $t[X] = t'[X]$, then $t[Y] = t'[Y]$, and the fact $t'[X], t'[Y] \in r(S)_H$. In the following discussion, the knowledge of $r(S)_H$ is assumed publicly available.

That is, for a given query:

$$Q = \{A_1, ..., A_i \mid P_k\}$$

$$Q \cup X \longrightarrow Y \models A_1, ..., A_i, ..., A_j$$

where $A_i \in X \subseteq \mathbb{A}_{PU}$, $P_k \in \mathbb{P}_{PU}$, $A_j \in Y \subseteq \mathbb{A}_{PR}$, and $1 \leq i \neq j \leq n$.

Given Armstrong Inference Axioms (IA) as follows [3],

IA1: Reflexive Rule

if $Y \subseteq X$, then $X \longrightarrow Y$.

IA2: Augmentation Rule

if $X \longrightarrow Y$, then $XZ \longrightarrow YZ$.

IA3: Transitive Rule

if $X \longrightarrow Y$ and $Y \longrightarrow Z$, then $X \longrightarrow Z$.

Some attribute values $V(\mathbb{A}_{PR})$ might be further inferred and disclosed by using some of IA, for example, **IA3**. The following will discuss the inference chain of the functional dependencies based on the closure and Armstrong Inference Axioms.

Definition 2.3 *A functional dependency $f: X \longrightarrow Y$ is trivial if $Y \subseteq X$. If $Y \not\subseteq X$, then functional dependency $f: X \longrightarrow Y$ is non-trivial.*

In the rest part, this paper refers to only set of non-trivial functional dependencies.

Definition 2.4 *Let F be a set of functional dependencies, and let $f \in F$ with respect to a set of attributes in f, $\sum(f) \in S$,*

$$F^+ = \{f \mid \sum(f) \subseteq \cup_{f' \in F} \sum(f') \wedge F \models_{\{Arms' rules\}} f\}$$

is called the **closure** *of F, derived by using Armstrong's rules* [3].

Definition 2.5 *For $X \subseteq S(A_1, ..., A_n)$ and a set of F of functional dependencies over $S(A_1, ..., A_n)$,*

$$closure_F(X) = \{Y \in S \mid X \longrightarrow Y \in F^+\}$$

is called **closure** *of X.*

For a given set of functional dependencies FDs,

$$F = \{f_i : X_i \longrightarrow Y_i \mid 1 \leq i \leq n\},$$

if $X_i \longrightarrow Y_i$ and $Y_i \longrightarrow Y_{i+1}$, then we have $X_i \longrightarrow Y_{i+1}$, denoted as $f_i \Rightarrow f_{i+1}$, by the transitivity property of Armstrong's rule[3].

Without loss of generality, if

$$f_1 \Rightarrow f_2, ..., f_i \Rightarrow f_{i+1}, ..., f_{n-1} \Rightarrow f_n,$$

denoted as,

$$f_1 \stackrel{*}{\Longrightarrow} f_n, \text{ then } X_i \longrightarrow Y_j, 1 \leq i < j \leq n.$$

For a given database query Q, although Q is allowed to access only the set of attributes $X_i \in \mathbb{A}_{PU}$, $1 \leq i \leq m$, if $f_1 \stackrel{*}{\Longrightarrow} f_n$, then $X_i \longrightarrow Y_j, 1 \leq i < j \leq n$, the value of $Y_j \subseteq \mathbb{A}_{PR}$ can be inferred, and will be disclosed. That is, for a given query:

$$Q = \{X_1, ..., X_i \mid P_k\}$$
$$Q \cup f_1 \stackrel{*}{\Longrightarrow} f_n \models X_1, ..., X_i, ..., Y_j$$

where $X_i \subseteq \mathbb{A}_{PU}$, $P_k \in \mathbb{P}_{PU}$, $Y_j \subseteq \mathbb{A}_{PR}$, and $1 \leq i \neq j \leq n$.

Given **IA2** and **IA3**, the pseudo transitive rule can be achieved:

IA4: Pseudotransitive Rule
if $X \longrightarrow Y$ and $YZ \longrightarrow W$, then $XZ \longrightarrow W$.

The proof of **IA4** can be done by following **IA2** and **IA3**[3].

For a given database query $Q \in \mathbb{Q}$ and access to attributes X, Y, Z, the value of W can be inferred and will be disclosed with the fact:

$$X \longrightarrow Y \ \& \ YZ \longrightarrow W \models XZ \longrightarrow W$$

That is, for given query:

$$Q = \{X, Y, Z \mid P_k\}$$
$$Q \cup (X \longrightarrow Y \& YZ \longrightarrow W) \models W$$

where $X, Y, Z \subseteq \mathbb{A}_{PU}$, $P_k \in \mathbb{P}_{PU}$, $W \subseteq \mathbb{A}_{PR}$.

2.2 General Inference Rules

For given set of various rules such as: $\mathbb{R} \subseteq \mathcal{KB}$, if there exists a rule $R : P_i \longrightarrow P_j$, where $\sum A_i \subseteq \mathbb{A}_{PU}$ is a set of attributes with respect to P_i and $A_j \in \mathbb{A}_{PR}$ is the attribute with respect to P_j. If $\sum A_i \subseteq \mathbb{A}_{PU}$ can be satisfied by a given query Q, then the value of $A_j \in \mathbb{A}_{PR}$ with respect to P_j can be inferred and will be disclosed.

That is, for a given query:

$$Q = \{\sum A_i \mid P_k\}$$

$$Q \cup (\mathbb{R} : P_i \longrightarrow P_j) \models A_j$$

where $\sum A_i \subseteq \mathbb{A}_{PU}$, $P_k \in \mathbb{P}_{PU}$, $\mathbb{R} : P_i \longrightarrow P_j \in \mathcal{KB}$, and $A_j \subseteq \mathbb{A}_{PR}$.

The type of each predicate P_i in R_i or a matched query condition in Q can be:

1. either a simple query predicate or constraint SQP such as:

$$A_i \ \theta \ C \text{ or } A_i \ \theta \ A_j, i \neq j,$$
$$\text{where } \theta \in \{=, <, \leq, >, \geq, \neq\}.$$

2. or a compound query predicate or constraint CQP that consists of a set of SQPs and logical operators such as: AND, OR, NOT.

For a given set of rules $\mathbb{R} \subseteq \mathcal{KB}$ such that:

$$\mathbb{R} = \{R_i : P_i \longrightarrow P_j \mid 1 \leq i = j + 1 \leq n\},$$

if $P_i \longrightarrow P_j$ and $P_{j+1} \longrightarrow P_k$, then we have $P_i \longrightarrow P_k$, denoted as $R_i \Longrightarrow R_{i+1}$, by the modus ponens law and hypothetical syllogism.

Without loss of generality, if

$$P_1 \longrightarrow P_2, ..., P_i \longrightarrow P_{i+1}, ..., P_n \longrightarrow P_{n+1},$$

denoted as, $P_1 \stackrel{*}{\longrightarrow} P_n$, then

$$R_i \stackrel{*}{\longrightarrow} R_j, 1 \leq i = j + 1 \leq n, \text{ or } R_1 \stackrel{*}{\Longrightarrow} R_n.$$

So for a given query:

$$Q = \{\sum A_i \mid P_k\}$$

$$Q \cup (\mathbb{R} : R_i \stackrel{*}{\Longrightarrow} R_j) \models A_j$$

where $\sum A_i \subseteq \mathbb{A}_{PU}$, $P_k \in \mathbb{P}_{PU}$, $\mathbb{R} : R_i \stackrel{*}{\Longrightarrow} R_j \subseteq \mathcal{KB}$, and $A_j \subseteq \mathbb{A}_{PR}$.

Definition 2.6 *Given predicates P_i, P_j, if $R_i : P_i \longrightarrow P_j$ and $R_j : P_j \longrightarrow P_i$, denoted as: $P_i \longleftrightarrow P_j$, or $P_j \longleftrightarrow P_i$, then $P_i \longleftrightarrow P_j$ is a mutual (circular) implication between P_i and P_j, or $R_i \Longleftrightarrow R_j$.*

Definition 2.7 *Given $R_i : P_i \longrightarrow P_{i+1}$ and $R_j : P_j \longrightarrow P_{j+1}$, $R_i, R_j \in \mathbb{R}$, $1 \leq i = j+1 \leq n$, if $P_i \stackrel{*}{\longrightarrow} P_j$ and $P_i \stackrel{*}{\longleftarrow} P_j$, denoted as $P_i \stackrel{*}{\longleftrightarrow} P_j$, or $P_i \stackrel{*}{\longrightarrow} P_j$ and $P_i \stackrel{*}{\longleftarrow} P_j$, denoted as $P_i \stackrel{*}{\longleftrightarrow} P_j$, then $R_i \stackrel{*}{\Longleftrightarrow} R_j$, $1 \leq i = j+1 \leq n$. We denote it as a mutual implication ring, or simply a ring.*

For given set of various rules $\mathbb{R} \subseteq \mathcal{KB}$ such that:

$$R_i \stackrel{*}{\Longleftrightarrow} R_j,\ 1 \leq i = j+1 \leq n.$$

if there exists a rule $R : P_i \stackrel{*}{\longleftrightarrow} P_j$, where $\sum A_i \subseteq \mathbb{A}_{PU}$ is a set of attributes with respect to P_i and $A_j \in \mathbb{A}_{PR}$ is the attribute with respect to P_j, and if $\sum A_i \subseteq \mathbb{A}_{PU}$ can be satisfied by a given query Q, then the value of $A_j \in \mathbb{A}_{PR}$ with respect to P_j can be inferred and will be disclosed.

So for a given query:

$$Q = \{\sum A_i \mid P_k\}$$

$$Q \cup (\mathbb{R} : R_i \stackrel{*}{\Longleftrightarrow} R_j) \models A_j$$

where $\sum A_i \subseteq \mathbb{A}_{PU}$, $P_k \in \mathbb{P}_{PU}$, $A_j \subseteq \mathbb{A}_{PR}$, and $\mathbb{R} : R_i \stackrel{*}{\Longleftrightarrow} R_j \subseteq \mathbb{R} \subseteq \mathcal{KB}$.

3 Discovering Inference Chains and Rings

Theorem 3.1 *For a given set of rules $\mathbb{R} \subseteq \mathcal{KB}$:*

$$\mathbb{R} = \{R_i : P_i \rightarrow P_j \mid 1 \leq i < j \leq n\}$$

with respect to a given schema $\mathbf{S}(A_1, ..., A_n)$, let

$$LHS_{\mathbb{R}} = \{P_i \mid P_i \in R_i, 1 \leq i \leq n\}$$

be the set of predicates on the left hand side of each $R_i \in \mathbb{R}$, and

$$RHS_{\mathbb{R}} = \{P_j \mid P_j \in R_i, 1 \leq i < j \leq n\}$$

be the set of attributes on the right hand,

a given set of rules $\mathbb{R} := \{P_i \longrightarrow P_j,\ 1 \leq i < j \leq n\}$ forms a mutual implication ring (MIR) for a given $\mathbb{R}$ set, if and only if $\mathbf{closure}(P_i) \subseteq (LHS_{\mathbb{R}} \cap RHS_{\mathbb{R}})$.

Proof:

Proof of the if-part:

Assume the condition $\mathbf{closure}(P_i) \subseteq (LHS_{\mathbb{R}} \cap RHS_{\mathbb{R}})$ is true, but $\mathbb{R} : P_i \stackrel{*}{\longleftrightarrow} P_j$ is not an MIR. For a given circular rule $\mathbb{R} : P_i \stackrel{*}{\longleftrightarrow} P_j$, both $P_i \rightarrow P_j$ and $P_i \rightarrow P_j$ must hold. Removal of either $P_i \longrightarrow P_j$ or $P_j \longrightarrow P_i$ will make $\mathbb{R} : P_i \stackrel{*}{\longleftrightarrow} P_j$ non-circular and contradict with the given condition $\mathbf{closure}(P_i) \subseteq (LHS_{\mathbb{R}} \cap RHS_{\mathbb{R}})$.

Proof of the only-if part: Assume $\mathbb{R} : P_i \stackrel{*}{\longleftrightarrow} P_i$ is an MIR, but the condition $\mathbf{closure}(P_i) \subseteq (LHS_{\mathbb{R}} \cap RHS_{\mathbb{R}})$ is not true. For a given circular rule $\mathbb{R} : P_i \stackrel{*}{\longleftrightarrow} P_j$, we have both $P_i \longrightarrow P_j$ and $P_j \longrightarrow P_i$. In order to make $P_i \subseteq LHS_{\mathbb{R}}$ true, but $P_i \not\subseteq RHS_{\mathbb{R}}$, removal of P_i on RHS implies the removal of $P_j \rightarrow P_i$, and it contradicts with the fact that $\mathbb{R} : P_i \longleftrightarrow P_j$ is an MIR as well as $\mathbb{R} : P_i \stackrel{*}{\longleftrightarrow} P_j$.

Algorithm 3.1 takes as the input: a given query Q, a given query predicate P_i, and a set of rules $\mathbb{R}$ with respect to schema $S(A_1, A_2, ..., A_n)$. The algorithm computes an inference chain $P_i \stackrel{*}{\longrightarrow} P_j$, $1 \leq i = j+1 \leq m$, for a given P_i with respect to a given query Q. The computation of an inference chain at Step 6c is based on the Armstrong Inference Axiom **IA3**, the transitive rule. Step 6g to 6l compute both the attributes $A_k \in \mathbb{A}_{PR}$ and $A_k \in \mathbb{A}$ implied by the predicates $P_j \in R_j$. Step 6m will test if the derived inference chain forms a ring $P_i \stackrel{*}{\longleftrightarrow} P_j$, $1 \leq i = j+1 \leq m$. If a ring $P_i \stackrel{*}{\longleftrightarrow} P_j$, $1 \leq i = j+1 \leq m$, is found, then the access to all the attributes in $A(P_i)$, $1 \leq i \leq m$, will be restricted in order to prevent any attribute values in the ring from disclosure. If a chain $P_i \stackrel{*}{\longrightarrow} P_j$, $1 \leq i = j+1 \leq m$ is found, then the attribute $A_{j-1} \in \mathbb{A}_{PU}$ will be restricted for the attribute $A_j \in \mathbb{A}_{PR}$ with respect to the inference chain $P_i \stackrel{*}{\longrightarrow} P_j$, $1 \leq i = j+1 \leq m$.

Algorithm 3.2 uses **Algorithm** 3.1 to discover all the inference chains and/or mutual implication rings. The

Algorithm 3.1 *Computing Chain of P_i*

Input: $P_i, Q,$ *and* $\mathbb{R}$ *with respect to* $S(A_1, ..., A_n)$
Output: **closure**(P_i), **chainclosure**(P_i), $\mathbf{A}(P_i)$, $\mathbf{A}(P_i)_{PR}$

begin

1. **oldclosure**$(P_i) := \emptyset$;
2. **newclosure**$(P_i) := \{P_i\}$;
3. **chainclosure**$(P_i) := \emptyset$;
4. $\mathbf{A}(P_i) := \emptyset$; $\mathbf{A}(P_i)_{PR} := \emptyset$;
5. **while** **oldclosure**$(P_i) \neq$ **newclosure**(P_i) **do**
6. **begin**
   (a) **oldclosure**$(P_i) :=$ **newclosure**(P_i);
   (b) **for** *each* $R_j : P_j \rightarrow P_k \in \mathbb{R}$ **do**
   (c) **if** $(P_j \subseteq$ **newclosure**$(P_i))$ **then**
   (d) **begin**
   (e) **newclosure**$(P_i) := \{P_k\} \cup$ **newclosure**(P_i);
   (f) **chainclosure**$(P_i) := \{P_j \rightarrow P_k\} \cup$ **chainclosure**(P_i);
   (g) **if** $R_j : P_j \rightarrow P_k \in \mathbb{R} \models A_k$ **then**
   (h) **begin**
   (i) $\mathbf{A}(P_i) := \mathbf{A}(P_i) \cup A_k$;
   (j) **if** $A_k \in \mathbb{A}_{PR}$ **then**
   (k) $\mathbf{A}(P_i)_{PR} := \mathbf{A}(P_i)_{PR} \cup A_k$;
   (l) **end**
   (m) **if** $(P_i == P_k)$ **then**
   (n) P_i.**count** := P_i.**count** +1;
   (o) **end**
7. **end**
8. **closure**(P_i) :=**newclosure**(P_i);
9. **return closure**(P_i), **chainclosure**(P_i);
10. **return** $\mathbf{A}(P_i)$, **return**$\mathbf{A}(P_i)_{PR}$;

end

Figure 1: Algorithm to Computing Chain of P_i

Algorithm 3.2 *Find Implication Chains and Rings*

Input: $S(A_1, ..., A_n)$, $Q, \mathbb{R}$
Output: set_{MIR}

begin

1. $set_{MIR} := \emptyset$; $set_{CH} := \emptyset$;
2. **for** *each* $R_i : P_i \rightarrow P_j \in \mathbb{R}$ **do**
3. **begin**
   (a) P_i.**count** := 0; P_i.**order** := 0;
   (b) *compute* **closure**(P_i) *by Algorithm 3.1*;
   (c) **if** $(P_i \subseteq LHS_{\mathbb{R}} \cap RHS_{\mathbb{R}})$ **then**
   (d) $set_{MIR} := set_{MIR} \cup$ **chainclosure**(P_i);
   (e) **else**
   (f) $set_{CH} := set_{CH} \cup$ **chainclosure**(P_i);

end

end

Figure 2: Algorithm to Find Implication Chains and Rings

results achieved from **Algorithm** 3.2 will be used for the corresponding prevention actions in **Trigger** 4.1 to be presented in Section 4.

4 The Active Approach

For a given active model based system such as:

$$\mathcal{T} = \{\mathcal{E}, \mathcal{C}, \mathcal{KB}, \mathcal{AC}\}$$

$\mathcal{AC}$ is a set of actions and defined as:

$$Q \times \mathcal{C} \times \mathcal{KB} \longmapsto \mathcal{AC}$$

where $\mathcal{AC} = \{\mathbb{AC}_b, \mathbb{AC}_p\}$, $\mathbb{AC}_b$ is the set of prior actions before an event $Q \in \mathbb{Q} \subseteq \mathcal{E}$ can proceed, and $\mathbb{AC}_p$ set of post actions after $Q \in \mathbb{Q} \subseteq \mathcal{E}$ is processed.

4.1 The Before Action $\mathbb{AC}_b$

When a query Q is issued, the active model $\mathcal{T}$ will take a proactive action $AC \in \mathbb{AC}_b$ before query Q can be processed. $AC \in \mathbb{AC}_b$ will invoke Algorithm **Algorithm** 3.2 to detect the inference mutual inference rings and chains. If a chain of inference rules is found such as:

$$P_1 \xrightarrow{*} P_n \models A_j \in \mathbb{A}_{PR},$$

where $1 \leq i = j - 1 \leq n$,

then the access to $A_i \in \mathbb{A}$, $A_i \sqsubseteq P_i \in R_i$ will be restricted in order to preserve the privacy of $\mathbb{A}_{PR}$.

If a ring of inference is found such as:

$$P_1 \xleftrightarrow{*} P_n \models A_j \in \mathbb{A}_{PR},$$

where $1 \leq i < j \leq n$,

then the access to all the attributes $A_i \in \mathbb{A}$, $A_i \sqsubseteq P_i$ will be restricted.

The active trigger **Trigger** 4.1 in **Figure** 3 will take the initial action to check the given query Q at Step 3a, and process query Q correspondingly at Step 3b, Step 3c, and Step 3d.

Trigger 4.1 *Checking Possible Disclosure*

Input: $S(A_1, ..., A_n)$, $Q, \mathbb{R}$
Output: set_{MIR}

begin

1. **EVENT:** Q *on database* $r(S)$
2. **CONDITION:** *if* $\exists A_i \in \mathbb{A}_{PR} \subseteq S(A_1, ..., A_n)$, $1 \leq i \leq n$
3. **ACTION:**
   (a) *invoke Algorithm 3.2 to detect* $P_i \xrightarrow{*} P_j$ *and* $P_i \xleftrightarrow{*} P_j, 1 \leq i < j \leq n$;
   (b) **if** $P_i \xrightarrow{*} P_j$ & $\models A_j \in \mathbb{A}_{PR}$ **then** *restrict the access to* $A_i \in \mathbb{A}$, $A_i \sqsubseteq P_i$; (*where* $i = j - 1$)
   (c) **if** $P_i \xleftrightarrow{*} P_j$ & $\models A_j \in \mathbb{A}_{PR}$ **then** *restrict the access to* $A_i \in \mathbb{A} \sqsubseteq P_i, \forall 1 \leq i \leq n$;
   (d) **if** $\forall A_i \sqsubseteq Q$ *are resstricted,* **then** *reject* Q;
4. *process* Q *with restriction on* A_i *in 3a and 3b*;

end

Figure 3: Disclosure Detection

4.2 The Post Action $\mathbb{AC}_p$

There are many actions in $\mathbb{AC}_p$ that can be taken after a query passes the testing in $\mathbb{AC}_b$. One of major $\mathbb{AC}_p$ is to keep track of query patterns by various data mining approaches. It is important to discover the association and correlations between $\mathbb{A}_{PU}$ and $\mathbb{A}_{PR}$ in the set of observed queries [1]. The higher ratio of association and correlations between $\mathbb{A}_{PU}$ and $\mathbb{A}_{PR}$ should trigger great attention and further actions. Mining query patterns is a broad subject area and will be considered for the future research.

5 Conclusions and Discussions

This paper has demonstrated an active model to detect possible disclosure of sensitive and private attribute values. The active approach mainly addresses a single rule inference, a chain type rule inference, and a ring type rule inference with respect to the type of functional dependency rules and generalized rules from different perspectives. The proposed algorithms detects the chains and rings, and the corresponding actions taken will limit the access to the private attribute values. Further, the post process will use data mining approach to find the patterns of query access for the future access control policy making.

6 Acknowledgments

The author would like to express sincere thanks to Dean and Dr. Edward Lieblein, faculty, and staff of Graduate School of Computer and Information Sciences, Nova Southeastern University.

References

[1] R. Agrawal, A. Evfimievski, J. Keirnan, and R. Velu, "Auditing Disclosure by Relevance Ranking," in *Proceedings of 2007 ACM SIGMOD International Conference on Management of Data*, pp. 79-90, 2007.

[2] R. Agrawal, J. Keirnan, R. Srikant, and Y. R. Xu, "Hippocratic Databases," in *Proceedings of 28th International Conference on Very Large Data Bases*, Hong Kong, China, pp. 143-154, 2002.

[3] W. W. Armstrong, "Dependency Structures of Data Base Relationships," in *Proceedings of IFIP Congress*, pp. 580-583, 1974.

[4] E. Bertino and R. Sandhu, "Database Security - Concepts, Approaches, and Chanllenges," in *IEEE Transactions on Dependable and Secure Computing*, Vol. 2, No. 1, pp. 2-19, 2005.

[5] S. Castano, M. Fugini, G. Martella, and P. Samarati. *Database Security* (Eds), ACM Press, 1994.

[6] K. LeFevre, R. Agrawal, V. Ercegovac, R. Ramakrishnan, Y. R. Xu, D. DeWitt, "Limiting Disclosure in Hippocratic Databases," in *Proceedings of* 30^{th} *International Conference on Very Large Data Bases*, Toronto, Canada, pp. 108-119, 2004.

[7] P. Stahlberg, G. Miklau, and B. N. Levine, "Threats to Privacy in the Forensic Analysis of Database Systems," *Proceedings of 2007 ACM SIGMOD International Conference on Management of Data*, pp. 91-102, 2007.