ISSN:2231-0711
www.ijcset.net



# Privacy Preserving and Security Control Method for Statistical Database

S.Udhaya Kumar and N.Partheeban

*Dept. of Computer Science & Engineering,*
*S.A. Engineering College,Anna University , Chennai, India.*

udhayakumarsoft@gmail.com

***Abstract*---As organizations increase their reliance on, possibly distributed, information systems for daily business, they become more vulnerable to security breaches even as they gain productivity and efficiency advantages. The availability of huge numbers of databases recording a large variety of information about individuals makes it possible to discover information about specific individuals by simply correlating all the available databases. Suppose a person AA owns a k-anonymous database and needs to determine whether her database, when inserted with a tuple owned by other person BB, is still k-anonymous. A release is considered k-anonymous if the information for each person contained in the release cannot be distinguished from at least k − 1 other persons whose information also appears in the release. Also, if the access to the database is strictly controlled, for example data are used for certain experiments that need to be maintained confidential. Clearly, allowing AA to directly read the contents of the tuple breaks the privacy of BB (e.g., a patient's medical record) on the other hand, the confidentiality of the database managed by AA violated once BB has access to the contents of the database. Thus, the problem is to check whether the database inserted with the tuple is still k-anonymous, without letting the two persons AA and BB know the contents of the tuple and the database, respectively. In this paper, we propose two protocols solving this problem on suppression-based and generalization-based k-anonymous and confidential databases. The protocols rely on well-known cryptographic assumptions, and we provide theoretical analyses to proof their soundness and experimental results to illustrate their efficiency.**

***Keywords*---Privacy, Anonymity, Data Management, Secure Computation, Data Confidentiality.**

## I. INTRODUCTION

DATA confidentiality is particularly relevant because of the value, often not only monetary, that data have. For example, medical data collected by following the history of patients over several years may represent an invaluable asset that needs to be adequately protected. Such a requirement has motivated a large variety of approaches aiming at better protecting data confidentiality and data ownership. Relevant approaches include query processing techniques for encrypted data and data watermarking techniques. Data confidentiality is not, however, the only requirement that needs to be addressed. Today here is an increased concern for privacy. The availability of huge numbers of databases recording a large variety of information about individuals makes it possible to discover information about specific individual's bysimply correlating all the available databases. Although confidentiality and privacy are often used as synonyms, they are different concepts: data confidentiality is about the difficulty (or impossibility) by an unauthorized user to learn anything about data stored in the database. Usually, confidentiality is achieved by enforcing an access policy, or possibly by using some cryptographic tools. Privacy relates to what data can be safely disclosed without leaking sensitive information regarding the legitimate owner [5]. Thus, if one asks whether confidentiality is still required once data have been anonymized, the reply is yes if the anonymous data have a business value for the party owning them or the unauthorized disclosure of such anonymous data may damage the party owning the data or other parties. The term anonymized or anonymization means identifying information is removed from the original data to protect personal or private information. There are many ways to perform data anonymization. We only focus on the k-anonymization approach.

As organizations increase their reliance on, possibly distributed, information systems for daily business, they become more vulnerable to security breaches even as they gain productivity and efficiency advantages. Though a number of techniques, such as encryption and electronic signatures, are currently available to protect data when transmitted across sites, a truly comprehensive approach for data protection must also include mechanisms for enforcing access control policies based on data contents, subject qualifications and characteristics, and other relevantcontextual information, such as time. Over the certain years the database security community has developed a number of different techniques and approaches to assure data confidentiality, integrity, and availability. However, despite such advances, the database security area faces several new challenges. Factors such as the evolution of security concerns, the "disintermediation" of access to data, new computing paradigms and applications, such as grid-based computing and on demand business, have introduced both new security requirements and new contexts in which to apply and possibly extend current approaches.

## II. RELATED WORK

The earlier protocols have some serious limitations, in that they do not support generalization-based updates, which is the main strategy adopted for data anonymization.

The first research direction deals with algorithms for database anonymization. The idea of protecting databases through data suppression or data perturbation has been extensively investigated in the area of statistical database. The problem of computing a k-anonymization of a set of tuples while maintaining the confidentiality of their content.

The second research direction is related to Secure Multiparty Computation (SMC) techniques. SMC represents an important class of techniques widely investigated in the area of cryptography. However, these techniques generally are not efficient.

The third research direction is related to the area of private information retrieval, which can be seen as an application of the secure multiparty computation techniques to the area of data management. The problem of privately updating a database has not been addressed in that these techniques only deal with data retrieval.

Finally, the fourth research direction is related to query processing techniques for encrypted data. The approaches do not address the k-anonymity problem since their goal is to encrypt data, so that their management can be outsourced to external entities.

## III. BASIC DEFINITIONS AND PRIMITIVES

### A. *Anonymity Definitions*

We consider a table T {T1,…..Tn} over the attribute set A. The idea is to form subsets of indistinguishable tuples by masking the values of some well-chosen attributes. In particular, when using a suppression-based anonymization method, we mask with the special value, the value deployed for the anonymization. When using a generalization-based anonymization method, original values are replaced by more general ones, according to a priori established value generalization hierarchies.

### B. *Cryptographic Primitives*

A commutative, product-homomorphic encryption scheme ensures that the order in which encryptions are performed is irrelevant (commutativity) and it allows to consistently perform arithmetic operations over encrypted data We extend the definition of commutative, indistinguishable encryption scheme presented in , in order to obtain an encryption scheme which also product-homomorphic.

## IV. PRIVACY-PRESERVING DATA MANAGEMENT TECHNIQUES

Data represent an important asset. We see an increasing number of organizations that collect data, often concerning individuals, and use them for various purposes, ranging from scientific research, as in the case of medical data, todemographic trend analysis and marketing purposes. Organizations may also give access to the data they own or even release such data to third parties. The number of increased data sets that are thus available poses serious threatsagainst the privacy of individuals and organizationsparties. In this case, data once are released are no longer under the control of the organizations owning them. Therefore, the organizations that are owners of the data are not able to control the way data are used.

Privacy is an important concern, several research efforts have been devoted to address issues related to the development of privacy-preserving data management techniques. A first important class of techniques deals with privacy preservation when data are to be released to thirdparties. In this case, data once are released are no longer under the control of the organizations owning them. Therefore, the organizations that are owners of the data are not able to control the way data are used. The most common approach to address the privacy of released data is to modify the data by removing all information that can directly link data items with individuals; such a process is referred to as data anonymization.

## V. THE PROTOCOLS

### A. *Private Update for Suppression-based Anonymous and Confidential Databases*

Here, we assume that the database is anonymized using a suppression-based method. Here our protocols are not required to further improve the privacy of users other than that provided by the fact that the updated database is still k-anonymous. Suppose that AA owns a k-anonymous table T over the QI attributes. AA has to decide whether T [ t—where t is a tuple owned by BB—is still k-anonymous, without directly knowing the values in t (assuming t and T have the same schema). This problem amounts to decide whether t matches any tuple in T on the nonsuppressed QI attributes. If this is the case, then t, properly anonymized, can be inserted into T. Otherwise, the insertion of t into T is rejected.

A trivial solution requires as a first step AA to send BB the suppressed attributes names, for every tuple in the witness set of T. In this way, BB knows what values are to be suppressed from his tuple. After BB computes the anonymized or suppressed versions of tuple t, AA can start a protocol (e.g., the Intersection Size Protocol) for privately testing the equality of _tiand _i. As a drawback, BB gains knowledge about the suppressed attributes of AA. A solution that addresses such drawback is based on the following protocol. Assume, AA and BB agrees on acommutative and product-homomorphic encryption scheme E. Further, they agree on a coding as well. Since other non-QI attributes do not play any role in our computation, without loss of generality, let the tuple contains only thenonsuppressed QI attributes of witness. The Protocol allows AA to compute an anonymized version of t withoutBB know what are the suppressed attributes of the tuples in T.

The protocol works as follows:

The step 1 starts with AA sends BB an encrypted version, containing only the s nonsuppressed QI attributes. At Step 2, BB encrypts the information received from AA and sends it to her, along with encrypted version of each value in his tuple t. At Steps 3-4, AA examines if the nonsuppressed QI attributes i is equal to those of t.

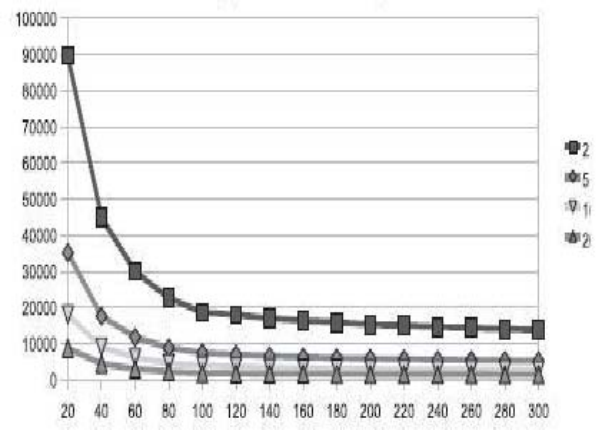



**Fig 1: Suppresion-Based Update**

*B. Private Update for Generalization-based Anonymous and Confidential Databases*

In this section, we assume that the table T is anonymized using a generalization-based method; let T1,… Tn be disjoint VGHs corresponding to A1…An2 Aanont known to AA. This can be safely performed without breaking the k-anonymity property. We will prove this claim later in the section. The protocol's details follow:

1. AA randomly chooses a value which returns specific value to each attribute.
2. AA computes the secure protocol.
3. AA and Bob collaboratively compute the secure protocol.
4. If s=u then t's generalized form can be safely inserted to T.
5. Otherwise, AA computes Tw Tw_ f_g and repeatthe above procedures until either s =u.

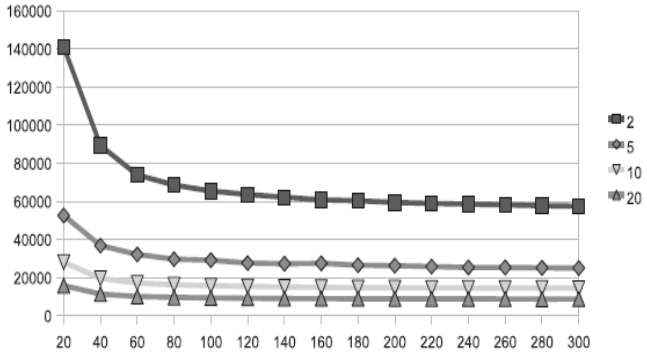



**Fig 2: Generalisation Based Update**

*C. Security Analysis*

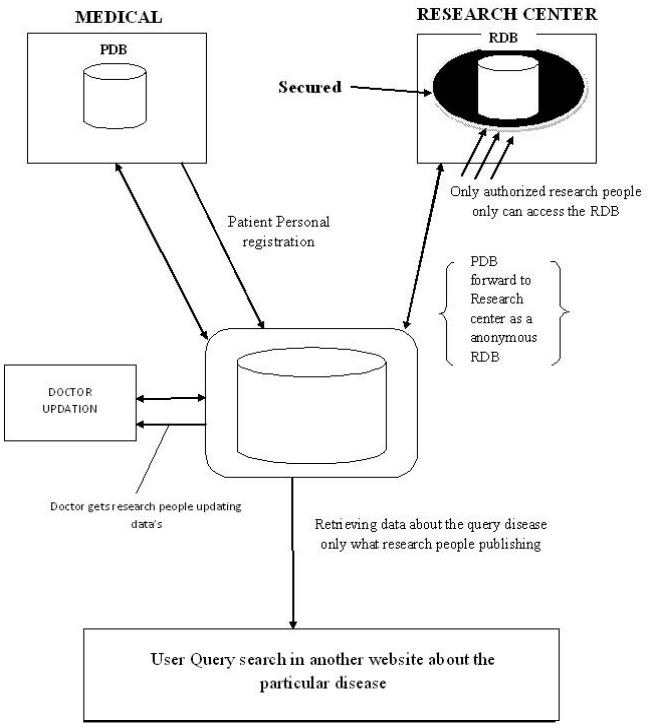
The security of Protocol 5.1 depends on that of the secure protocol (SSI), and detailed security analyses of SSI can be determined. The SSI protocol presented is easy to implement and efficient to perform. Although the protocol leaks the intersection size between andto the participating parties, it does provide sufficient privacy protection in our application. In case this linkage of intersection sizes is not acceptable, we can adopt one variation of the SSI protocol. We can make the protocol only return whether or not without disclosing the intersection size. Under the context of Secure Multiparty Computation, this variation of SSI does not leak any information that cannot be inferred from the final result and the private input data. Thus, using SSI Protocol 5.1 can achieve very high security.

## VI. ARCHITECTURE AND EXPERIMENTAL RESULTS

Our prototype of a Private Checker (that is, AA) is composed by the following modules: a crypto module that is in charge of encrypting all the tuples exchanged between an user (that is, BB) and the Private Updater, using the techniques exposed in Sections 4 and 5; a checker module that performs all the controls, as prescribed by earlier Protocols; a loader module that reads chunks of anonymized tuples from the k-anonymous DB. The chunk size is fixed in order to minimize the network overload. The modules are represented along with labelled arrows denoting what information is exchanged among them.

If none of the tuples in the chunk matches the User tuple, then the loader reads another chunk of tuples from the k-anonymous DB. Note the communication between the prototype and User is mediated by an anonymizer (like Crowds, not shown in figure) and that all the tuples are encrypted.

**SYSTEM ARCHITECTURE:**



## VII. EXPERIMENTAL RESULTS

**Methodologies**

Methodologies is the process of analyzing the principles or procedure of a Progressive Anonymous Database management system.

The following are the 7 modules involves in Anonymous database system.

MAIN MODULE'S:

- PATIENT ACCESS IMPLEMENTATION.
- DOCTOR TREATMENT ALLOCATION.
- MEDICAL UPDATION AND MONITORING.
- RESEARCH DATABASE ALLOCATION AND UPDATION.
- RESERCH PEOPLE RESEARCH UPDATION
- USER SEARCH.
- MEDICAL SEARCH

**Patient Access Implementation:**

This is the first step what patient to do. In this module patient want to register the personal details in the medical database and get the authentication processes to go forward. In this module patient want to give the database to medical admin all the registration process are done by a medical admin. After the registration process completed patient can get the authentication code and machine generated patient id, by using this only patient can login to the medical.

The sub modules's in the Patient Access Implementation comprises the following

- DOCTOR APPOINTMENT
- REQUESTS
- MY TREATMENTS


Fig: Medical Main Page

**Doctor Treatment Allocation:**

In this module authorized doctors can login into the medical. Here also all the details about the doctor are registered by the medical admin. And the medical admin give the authentication details to the particular doctor after getting the authentication details doctor can login to the medical and can start the below processes.

The submodules in the Doctor treatment Allocation comprises the following:

- PATIENT DETAILS
- APPOINTMENTS
- MY PATIENTS

Fig : Patient Get Doctor Appointment Page


**Fig : Doctor Appointment Page**

**MEDICAL UPDATION AND MONITORING:**

This module is only for administrator of the medical.

The submodules in the Medical Updation and Monitoring

- PATIENT DETAILS
- DOCTORS DETAILS
- NEW DOCTOR REGISTRATION
- NEW PATIENT REGISTRATION
- QUERIES


Fig : New Patient Registeration

**Research Database Allocation and Updation:**

In this module research people can see the data's send by medical. And allocate research peoples to each research data. And forward the data to research people. Here also research people can't do any changes or modifications in patient database they only can use the database for reference purpose.


Fig: Admin Login Page

**Research People Research Updation:**

In this module research people can get the allocated data's from the research admin. And research people can update research database what they find newly. The data published to the user or medical based on the research people updating, here we have to publish the database to user only the specific data.


Fig: User Query

**USER SEARCH:**

In this module any user can get relevant information about the disease from the web which is newly updated by the research people.

**MEDICAL SEARCH:**

In this module medical can see the research people newly updated data's from research database.

## VIII. CONCLUSION/FUTURE WORK

In this paper, we have presented two secure protocols for privately checking whether a k-anonymous database retainsits anonymity once a new tuple is being inserted to it. Since the proposed protocols ensure the updated database remainsk-anonymous, the results returned from a user's (or a medical researcher's) query are also k-anonymous. Thus, the patient or the data provider's privacy cannot be violated from any query. As long as the database is updated properly. Using the proposed protocols, the user queries under our application domain are always privacy-preserving.

In order for a database system to effectively perform privacy preserving updates to a k-anonymous table, other important issues are to be addressed:

a. The definition of a mechanism for actually performing the update, once k-anonymity has been verified;
b. The specification of the actions to take in yield a negative answer;
c. How to initially populate an empty table;
d. The integration with a privacy-preserving query system.

Here, we sketch the solutions developed in order to address these questions and which comprise our overall methodology for the private database update. As a general approach, we separate the process of database k-anonymity checking and the actual update into two different phases, managed by two different subsystems: the Private Checker and the Private Updater. In the first phase, the Private Checker prototype, checks whether the updated database is still k-anonymous, without knowing

The content of the user's tuple. In the second phase, the Private Updater actually updates the database based on theresult of the anonymity check; we refer to this step as update execution.

Then, the system does not insert the tuple to the k-anonymous database, and waits until k _ 1 other tuples fail the insertion. At this point, the system checks whether such set of tuples, referred to as pending tuple set, are k-anonymous In addition to the problem of falling insertion, there are other interesting and relevant issues that remain to beaddressed:

- Devising private update techniques to database systems that support notions of anonymity different than k-anonymity.
- Dealing with the case of malicious parties by the introduction of an untrusted, non colluding third party [12].
- Implementing a real-world anonymous database system.
- Improving the efficiency of protocols, in terms of number of messages exchanged and in terms of their sizes, as well.

## ACKNOWLEDGMENT

I must thank, first and foremost, my Coordinator MR.S. Muthukumarasamy M.E and Internal Guide Mr.N. Partheeban. M.E.(Ph.D)., Assistant Professor, Department of Computer Science and Engineering, without whose guidance and patience, this dissertation would not be possible. I wish to record my thanks to Mrs. Umarani Srikanth M.E., (Ph.D) Head of the Department, Department of Computer Science and Engineering, project panel members, Professors of the Department of Computer Science and Engineering for their consistent encouragement and ideas.

## REFERENCES

[1] Alberto Trombetta, Wei Jiang, Member, IEEE, Elisa Bertino, Fellow, IEEE, and Lorenzo Bossi, "Privacy-Preserving Updates to Anonymous and Confidential Databases", july 2011

[2] N.R. Adam and J.C. Wortmann, "Security-Control Methods for statistical Databases: A Comparative Study," 1989.

[3] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy,D. Thomas, and A. Zhu, "Anonymizing Tables," 2005.

[4] R. Agrawal, A. Evfimievski, and R. Srikant, "Information Sharingacross Private Databases," 2003.

[5] C. Blake and C. Merz, "UCI Repository of Machine Learning Databases," 1998.

[6] E. Bertino and R. Sandhu, "Database Security—Concepts, Approaches and Challenges," Mar. 2005.

[7] D. Boneh, "The Decision Diffie-Hellman Problem," 1998.

[8] D. Boneh, G. di Crescenzo, R. Ostrowsky and G.Persiano, "PublicKey Encryption with Keyword search," 2004.

[9] S. Brands, "Untraceable Offline Cash in Wallets with Observers," 1994.

[10] J.W. Byun, T. Li, E. Bertino, N. Li, and Y. Sohn, "Privacy-Preserving Incremental Data Dissemination," 2009.

[11] R. Canetti, Y. Ishai, R. Kumar, M.K. Reiter, R. Rubinfeld, and R.N.Wright, "Selective Private Function Evaluation with Applicationto Private Statistics," 2001.

[12] S. Chawla, C. Dwork, F. McSherry, A. Smith, and H. Wee, "Towards Privacy in Public Databases,", 2005.