ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

Data Management By Privacy Preserving In The Anonymous Database Using Suppression and Generalization protocols

J Yashwanth kumar[1], G Kalpana[2]

M.Tech-SE, DRK Institute of Science and Technology, JNTUH, Hyderabad, India[1]

ASSOCIATE PROFFESOR, DRK Institute of Science and Technology, JNTUH, Hyderabad, India[2]

ABSTRACT: The main aim of our project is to give confidentiality to user details in a database and we are going to show this by using a medical process and that medical DB. When a patient newly comes to a medical his information must be collected so here also we are going to do the same thing. The medical database admin (MDA) will store all the fresh information about that patient in the anonymous database. And give one user name id password to that patient to access his/her page created by that medical. When user wants to open his/her page, he/she can use that user id and password that page. In that page all doctor information will be available who all are working in that medical. By that list patient can give request to the preference doctor. then request will go to the particular doctor page. If doctor approve the particular patient appointment request. Then the patient add as his/her OWN PATIENT list and can start the treatment. Dr can store all the patient information (Treatment, medicine, status) in the anonymous database and when treatment would be finished Dr can use the complete action on his/her page then only all the information regarding that patient will to that anonymous database . Then after completion Dr will send the Patient id to the MDA. Here MDA will send approval request to the patient to send his/her details to the researchers and if patient will give the permission then only MDA will send the information to the researchers. After getting the details researcher admin will allocate maximum sub researcher to researching about that disease and treatment now the maximum sub researchers only can access the database for research and save all the researching information in anonymous database but in two ways one for user and another one for doctor. If user search about the disease in web. The user can get only limited information from the anonymous database. And also doctor also can get the particular new treatment and medical information from the anonymous database.

Keywords: Privacy, anonymity, data management, secure computation, confidentiality, cryptography.

I. INTRODUCTION

Today there 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 individuals by simply correlating all the available databases. Although confidentiality and privacy are often used as synonyms, they are different concepts: data confidentiality is about the difficulty 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 sensitive information regarding the legitimate owner.

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. To better understand the difference between confidentiality and anonymity, consider the case of a medical facility connected with a research institution. Suppose that all patients treated at the facility are asked before leaving the facility to donate their personal health care records and medical histories (under the condition that each patient's privacy is protected) to the research institution, which collects the records in a research database. To guarantee the maximum privacy to each patient, the medical facility only sends to the research database an anonymized version of the patient record. Once this anonymized record is stored in the research database, the nonanonymized version of the record is removed from the system of the medical facility. Thus, the research database used by the researchers is anonymous. Suppose that certain data concerning patients are related to the use of a drug over a period of four years and certain side effects have been observed

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

and recorded by the researchers in the research database. It is clear that these data (even if anonymized) need to be kept confidential and accessible only to the few researchers of the institution working on this project, until further evidence is found about the drug. If these anonymous data were to be disclosed, privacy of the patients would not be at risk; however the company manufacturing the drug may be adversely affected. Recently, techniques addressing the problem of privacy via data anonymization have been developed, thus making it more difficult to link sensitive information to specific individuals.

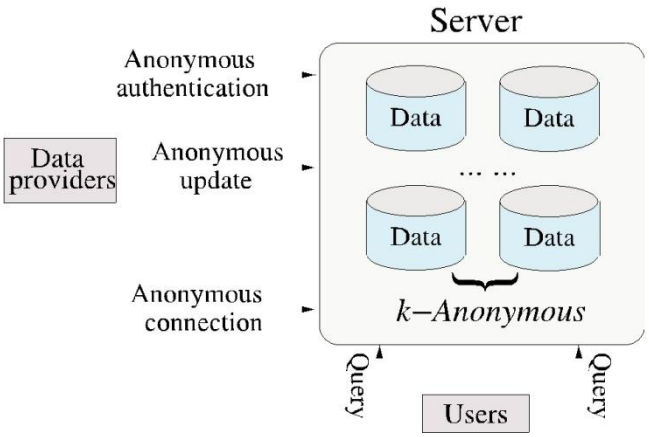

Fig. 1. Anonymous Database System.

One well-known technique is k-anonymization. Such technique protects privacy by modifying the data so that the probability of linking a given data value, for example a given disease, to a specific individual is very small. So far, the problems of data confidentiality and anonymization have been considered separately. However, a relevant problem arises when data stored in a confidential, anonymity-preserving database need to be updated. The operation of updating such a database, e.g., by inserting a tuple containing information about a given individual, introduces two problems concerning both the anonymity and confidentiality of the data stored in the database and the privacy of the individual to whom the data to be inserted are related: 1) Is the updated database still privacy-preserving? and 2) Does the database owner need to know the data to be inserted? Clearly, the two problems are related in the sense that they can be combined into the following problem: can the database owner decide if the updated database still preserves privacy of individuals without directly knowing the new data to be inserted? The answer we give in this work is affirmative.

It is important to note that assuring that a database maintains the privacy of individuals to whom data are referred is often of interest not only to these individuals, but also to the organization owning the database. Because of current regulations organizations collecting data about individuals are under the obligation of assuring individual privacy. It is thus, in their interest to check the data that are entered in their databases do not violate privacy, and to perform such verification without seeing any sensitive data of an individual.

I.i Existing Methodologies

In the existing system data are store in database directly. Anyone can easily retrieve information like username, password. Etc. The cryptography security is not maintained here. The classification of database is carried out from local system only. Any unauthorized person can easily access the database. Authorized person can view the other user's data too. Data confidentiality is particularly relevant because of the value, often not only monetary, that data possess. A requirement has motivated a large variety of approaches aiming at better protecting data confidentiality and data ownership. 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.

I.ii Proposed Methodologies

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 prove their soundness and experimental results to illustrate their efficiency. It is today well understood that databases represent an important asset for many applications and thus their security is crucial.

Recently, techniques addressing the problem of privacy via data anonymization have been developed, thus making it more difficult to link sensitive information to specific individuals. One well-known technique is k-anonymization. Cryptography technique is using secure data storing in server.

The protocols we propose to solve Problem 1 rely on the fact that the anonymity of database is not affected by inserting *t* if the information contained in t, properly

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

anonymized, is already contained in DB. Then, Problem 1 is equivalent to privately checking whether there is a match between (a properly anonymized version of) t and (at least) one tuple contained in DB. The first protocol is aimed at suppression-based anonymous databases, and it allows the owner of DB to properly anonymized the tuple t, without

gaining any useful knowledge on its contents and without having to send to t's owner newly generated data. To achieve such goal, the parties secure their messages by encrypting them. In order to perform the privacy-preserving verification of the database anonymity upon the insertion, the parties use a commutative and homomorphism encryption scheme. The second protocol is aimed at generalization-based anonymous databases, and it relies on a secure set intersection protocol, such as the one found in to support privacy-preserving updates on a generalization-based k-anonymous DB.

II.Architecture

The figure 2 shows the flow of steps followed in the system. It starts with authentication of user. Each user is provided with username and password registered in system already. There is a salt value authentication along with password. The authentication user has access to the database and system has particular access rights for each user. The
Anonymous database suppresses and generalizes the data
according to data value. The database can be accessed by research centers for gathering statistical data regarding particular medicines, the percentage of curable medicines. The internal or private information of the patients are not revealed to the research centre computation. The research people can see the data's send by the database according to its access right. And allocate research peoples to each research data. And forward the data to research people. Here research people can't do any changes or modifications in patient database they only can use the database for reference purpose.

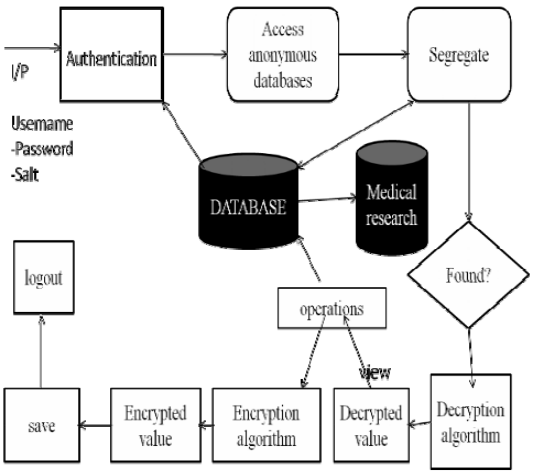

fig:2.1 Architecture

The authorized database updaters can login into the medical. Here also all the details about the database updater are registered by the admin. And the admin give the authentication details to the particular updater after getting the authentication details, can login to the database and can start the processes. The doctor and patient enter all details regarding their treatment details in the database in the hospital. These details are not disclosed to the research centre. The data can be encrypted and saved and can be decrypted back to original form when required. For example, in the proposed system, even the administrator has only restricted access to database, he can't access the internal details of each user, and rather he can find how many users are updated and solve issues regarding users. Individual users are not allowed access to other accounts except their personal record. Figure 3 shows view of the system for an example patient database where there is a medical database, where the patient registers their details initially. The doctor can view their necessary information about the patient and also fix appointments for patients easily. The patients can in turn fix appointments with their doctor in charge and reduces waiting time for patients due to appointments. The doctor can in turn update record of patients and their treatments to the patient database. Also the doctor can retrieve information from other sources regarding the illnesses and their treatment

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

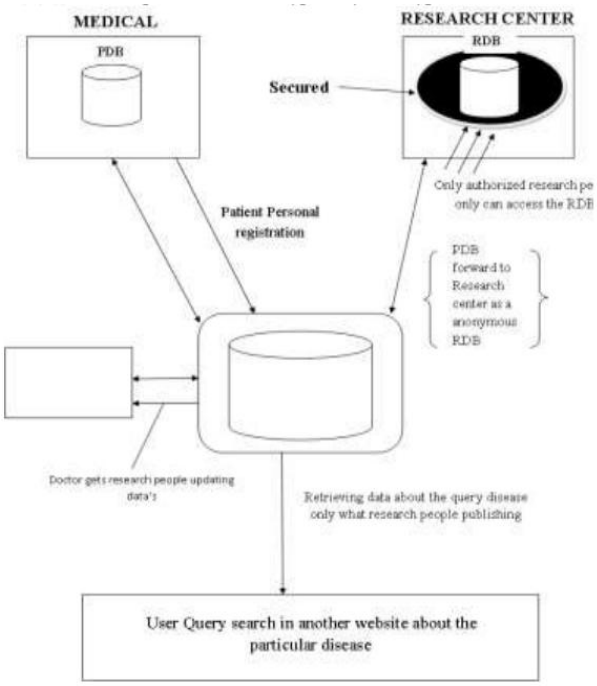

Fig:2.2 Architecture

III. A Private Update Protocol for Suppression-based Anonymous Databases

In this section, we assume that the database is anonymized using a suppression-based method. Note that 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 Alice owns a k-anonymous table T over the QI attributes. Alice has to decide whether T when inserted with a tuple t, owned by Bob, 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 non-suppressed Q I 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 Alice to send Bob the suppressed attributes names, for every tuple in the witness set (δ_1,. . . ,δs) of T. In this way, Bob knows what values are to be suppressed from his tuple. After Bob computes the anonymized or suppressed versions of tuple t, 1 5 i 5 w, he and Alice can start a protocol for privately testing the equality of 6 and Ji. As a drawback, Bob gains knowledge about the suppressed attributes of Alice. A solution that addresses such drawback is based on the following protocol. Assume Alice and Bob agree on a commutative and product-homomorphism encryption scheme E and Q I = {A1,. . . , A,). Further, they agree on a coding c(-) (Equation 4) as well. Since other non-QI attributes do not play any role in our computation, without loss of generality, let Ji = (v:, . . . , v:) be the tuple containing only the s non-suppressed Q I attributes of witness wi, and t = (vl, . . . , v,). Protocol 4.1 allows Alice to compute an anonymized version of t without letting her know the contents oft and, at the the same time, without letting Bob know what are the suppressed attributes of the tuples in T. The protocol works as follows: at Step 1, Alice sends Bob an encrypted version of Ji, containing only the s non-suppressed QI attributes. At Step 2, Bob encrypts the information received from Alice and sends it to her, along with encrypted version of each value in his tuple t. At Steps 3-4, Alice examines if the non-suppressed QI attributes of hi is equal to those of t.

Protocol III.i

1. Alice codes her tuple δ_i into $c((v_i',. . . , v_s'))$, denoted as $c(\delta_i)$. Then, she encrypts c(Ji) with her private key and sends $E_A(c(\delta_i))$ to Bob.
2. Bob individually codes each attribute value in t to get the tuple of coded values, $c((v_i',. . . , v_s'))$, encrypts each coding and $c(\delta_i)$ with his key B and sends $E_A(cv_1))\ldots E_A(cv_u))$ and (ii) $E_B(E_A(c(\delta_i)))$ to Alice.
3. Since E is a commutative scheme, $E_B(E_A(c(\delta_i)))= E_A(E_B(c(\delta_i)))$, Alice decrypts $E_B(E_A(c(\delta_i)))$ to obtain $E_B(c(\delta_i))$. Since the encrypted values sent by Bob are ordered according to the ordering of the attributes in T (assume this is a public information known to both Alice and Bob), Alice knows which are, among the encrypted values sent by Bob, the one corresponding to the suppressed and nonsuppressed QI attributes. Thus, Alice computes
   $E_B(c(v_1)).x\ldots.x\ E_B(c(v_s))$
   where v_1 . . . ; v_s are the values of nonsuppressed attributes contained in tuple t. As already mentioned,E is a product-homomorphism encryption scheme. Based also on the definition of function c(.) this implies that Expression 5 is equal to
   $E_B(c(<v_1\ldots..v_s)>)$.
4. Alice checks whether
   $E_B(c(<v_1\ldots..v_s)>)= E_B(c(<v_1^1\ldots..v_s^1)>)$.
   If true, t (properly anonymized) can be inserted to

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

table T. Otherwise, when inserted to T, t breaks
K-anonymity.

IV. A Private Update Protocol for Generalization-based Anonymous Databases

In this section, we assume that the table T is anonymized using a generalization-based method; let rl, . . . , I?, be u disjoint value generalization hierarchies (VGHs) corresponding to A1,. . . , A, E Atnon known to Alice. Let S E T, and let GetSpec(SIAl, . . . ,A,], rl, . . . , I',) (GetSpec(6) for short) denote a function which returns a set y of specific values (values at the bottom of a VGH) related to each attribute Ai E Atnon such that every value in y can be generalized to S[Ai] for some i according to ri. For example, let T refer to Table 4 and Atnon = {AREA, POSITION, SALARY). If S = [Operating Systems, Research Assistant, [llk,30k]], then based on the VGHs (presented in Figure 2) GetSpec(6) = {Distributed Systems, Handheld Systems, Research Assistant, $15,000,
$17,000, $15,500). Let t be Bob's private tuple, and assume that Bob knows the set AFnon. Bob can generate a set T containing the corresponding values t[A1], . . . , t[A,]; the size of T is always u. We denote by SSI(y, T) as a secure protocol that computes the cardinality of y n ~ . Such protocols can be found in [3, 121. Upon receiving an initial request from Bob, Alice starts the protocol by randomly choosing a tuple S from the witness set Tw of T. After Alice computes y = GetSpec(S), she and Bob privately compute SSI(y, 7). Note that Bob does not need to know any I'i. We claim that if SSI(y, T) = u (the size of A:non), t[A1,. . . ,A,] can be generalized to S, and hence this insertion into T can be safely performed without breaking the k-anonymity property.

Protocol IV.i.

1. Alice randomly chooses a δ Tw.
2. Alice computes γΓ GetSpec(δ)γ
3. Alice and Bob collaboratively compute s= SSI(γ,Γ)
4. If s =u then t's generalized form can be safely inserted
to T.
5. Otherwise, Alice computes Tw ←Tw –{δ}and repeat the above procedures until either s =u or Tw =Ø

V. CONCLUSIONS

In this paper, we have presented two secure protocols for privately checking whether a k-anonymous database retains its anonymity once a new tuple is being inserted to it.

Since the proposed protocols ensure the updated database remains k-anonymous, the results returned from 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, Protocols are necessary but clearly not sufficient. Other important issues are to be addressed:

1). The definition of a mechanism for actually performing the update, once k-anonymity has been verified;

2). the specification of the actions to take in case
Protocols 4.1 or 5.1 yield a negative answer;

3). how to initially populate an empty table; and

4). the integration with a privacy-preserving query
system.

5). Implement real world database system.

In the following, 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 presented in Section 6, following Protocol 4.1 or Protocol 5.1, 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 the result.

Concerning the actual execution of the database update, once the system has verified that the user's tuple can be safely inserted to the database without compromising k-anonymity, the user is required to send to the Private Updater the nonanonymous attributes' values to be stored in the k-anonymous database as well. The deployment of an

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

anonymity system ensures that the system cannot associate the sender of the tuple with the subject who made the corresponding insertion's request.

The important issues in future will be resolved:
1) Implement database for invalid entries.
2) Improving efficiency of protocol in terms of number of messages exchanged between user and database.
3) Implement real world database system.

ACKNOWLEDGMENT

I am grateful to my university which gave me opportunity and lots of resources to accomplish my project in successful way. Without the guidance of my faculties I would not have reached this far, I thank them all for their timely and scholarly suggestions and motivation.

REFERENCES

[1] N.R. Adam and J.C. Wortmann, "Security-Control Methods for Statistical Databases: A Comparative Study," ACM Computing Surveys, vol. 21, no. 4, pp. 515-556, 1989.
[2] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu, "Anonymizing Tables," Proc. Int'l Conf. Database Theory (ICDT), 2005.
[3] R. Agrawal, A. Evfimievski, and R. Srikant, "Information Sharing across Private Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2003.
[4] C. Blake and C. Merz, "UCI Repository of Machine LearningDatabases,"http://www.ics.uci.edu/mlearn/ML Repository. html, 1998.
[5] E. Bertino and R. Sandhu, "Database Security—Concepts, Approaches and Challenges," IEEE Trans. Dependable and Secure Computing, vol. 2, no. 1, pp. 2-19, Jan.-Mar. 2005.
[6] D. Boneh, "The Decision Diffie-Hellman Problem," Proc. Int'l Algorithmic Number Theory Symp., pp. 48-63, 1998.
[7] D. Boneh, G. di Crescenzo, R. Ostrowsky, and G. Persiano, "Public Key Encryption with Keyword Search," Proc. Eurocrypt Conf., 2004
[8] S. Brands, "Untraceable Offline Cash in Wallets with Observers," Proc. CRYPTO Int'l Conf., pp. 302-318, 1994.[9] J.W. Byun, T. Li, E. Bertino, N. Li, and Y. Sohn, "Privacy- Preserving Incremental Data Dissemination," J. Computer Security, vol. 17, no. 1, pp. 43-68, 2009.
[10] R. Canetti, Y. Ishai, R. Kumar, M.K. Reiter, R. Rubinfeld, and R.N. Wright, "Selective Private Function Evaluation with Application to Private Statistics," Proc. ACM Symp. Principles of Distributed Computing (PODC), 2001.
[11] S. Chawla, C. Dwork, F. McSherry, A. Smith, and H. Wee, "Towards Privacy in Public Databases," Proc. Theory of Cryptography Conf. (TCC), 2005.
[12] U. Feige, J. Kilian, and M. Naor, "A Minimal Model for Secure Computation," Proc. ACM Symp. Theory of Computing (STOC), 1994.
[13] M.J. Freedman, M. Naor, and B. Pinkas, "Efficient PrivateMatching and Set Intersection," Proc. Eurocrypt Conf., 2004.
[14] B.C.M. Fung, K. Wang, A.W.C. Fu, and J. Pei, "Anonymity for Continuous Data Publishing," Proc. Extending Database Technology Conf. (EDBT), 2008.
[15] O. Goldreich, Foundations of Cryptography: Basic Tools, vol. 1. Cambridge Univ. Press, 2001.
[16] O. Goldreich, Foundations of Cryptography: Basic Applications, vol. 2.Cambridge Univ. Press, 2004.
[17] H. Hacigu¨mu¨ s¸, B. Iyer, C. Li, and S. Mehrotra, "Executing SQL over Encrypted Data in the Database-Service-Provider Model," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2002.
[18] Y. Han, J. Pei, B. Jiang, Y. Tao, and Y. Jia, "Continuous Privacy Preserving Publishing of Data Streams," Proc. Extending Database Technology Conf. (EDBT), 2008.
[19] US Department of Health & Human Services, Office for Civil Rights, Summary of the HIPAA Privacy Rule, 2003.
[20] J. Li, N. Li, and W. Winsborough, "Policy-Hiding Access Control in Open Environment," Proc. ACM Conf. Computer and Comm.Security (CCS), 2005.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

BIOGRAPHY

J Yashwanth Kumar is a student of DRK Institute of science and Technology, Ranga Reddy, Andhra Pradesh, India. He has received B.Tech degree in Computer Science and Engineering and M.Tech Degree in Computer Science.

G.Kalpana is working as Associate Professor at DRK Institute of Science & Technology, Ranga Reddy, and Andhra Pradesh, India. She has received M.Tech Degree in Computer Science.