Security Through The Lens Of Privacy And Confidentiality Using K-Technique

Sheedhal Thomas, Shruthi Prabhakaran, Snehal Salunkhe, Pallavi Kakade, B.S.Khade

Abstract: Suppose Alice owns a k-anonymous database and needs to determine whether her database, when inserted with a tuple owned by Bob, is still k-anonymous. Also, suppose that access to the database is strictly controlled, because for example data areused for certain experiments that need to be maintained confidential. Clearly, allowing Alice to directly read the contents of the tuple breaks the privacy of Bob (e.g., a patient's medical record); on the other hand, the confidentiality of the database managed by Alice is violated once Bob 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 Alice and Bob 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.

Index Terms: Anonymization, Authentication, Confidentiality, Cryptography, Decryption, Encryption, k-technique ,Privacy, Security.

1 INTRODUCTION

Nowadays it is well understood that databases represent an important resource for many applications and thus providing security to database is extremely important .Data in database have its own unique value. For example in hospitals record of patients suffering from various diseases may be recorded. If the hospital wants to reveal information data to a research center or any other pharmaceutical company, it should not give any hint related to that patient to the research center. It can provide information in the form of statistics of the particular disease that how many patients are suffering from that disease. There are large number of databases that contain many confidential information such that people can access those data relating to various information from various databases. Access rights should be given to users and information should be disclosed on the extent of access rights provided to the user.

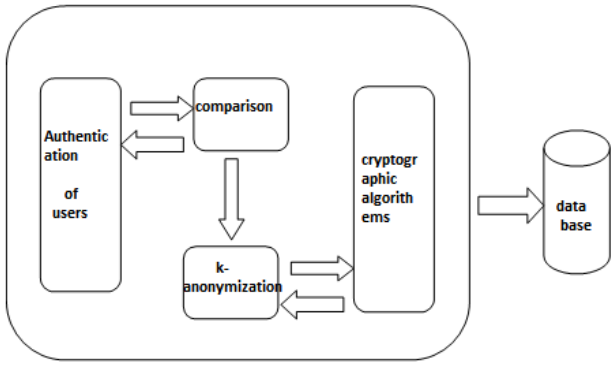

Fig -1: Basic Architecture

In today's world data confidentiality is not the only requirement, there is need for privacy too. Huge number of database recording contain large number of various information about individuals which makes it possible reveal information about specific individuals by just comparing all the related databases. Even though privacy and confidentiality have the same meaning they are different concepts. Confidentiality relates to hiding of information from unauthorized users. Privacy relates to protecting the private

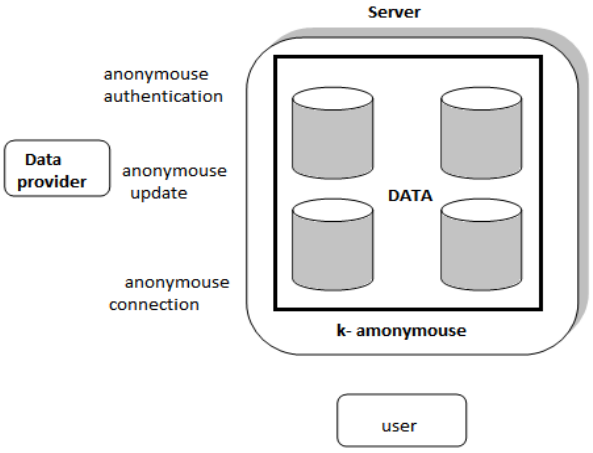

Fig -2: Application of K-Technique

information of the user It is important to assure that database maintains privacy not only of the individuals, but also the organization owing the database. Thus data entered in the database should be checked so that it does not violate the privacy.

PROBLEM STATEMENT

In figure1 lets assume information of a single patient is stored in a tuple and database is kept confidentially at the server. the user in figure 1 are treated as researchers. Database is contains sensitive data ,so privacy of the patient is the main concern. Suppose a new patient has to treated, the database has to be updated in order to store the medical data of the new patient. due to this entire database has to be revealed to the party managing the database server thus violating the privacy and confidentiality. Using client details as input, we aim at implementing two unique protocols that will enable our database to maintain its security, thus giving a confidential and privacy-preserved database. Thus our database will be a symphony of security.

PROPOSED METHODOLOGY

In this paper, we are using k-technique algorithm. The K-technique algorithm contains two protocols Suppression and Generalization. 1st protocol aims at suppressing data entered

in the database to allow the DB to properly anonymize the data. 2nd protocol aims at generalization-based anonymous databases to support privacy-preserving updates. Suppression:- replace attributes with * Generalization:- age=26 then age:[20-30]

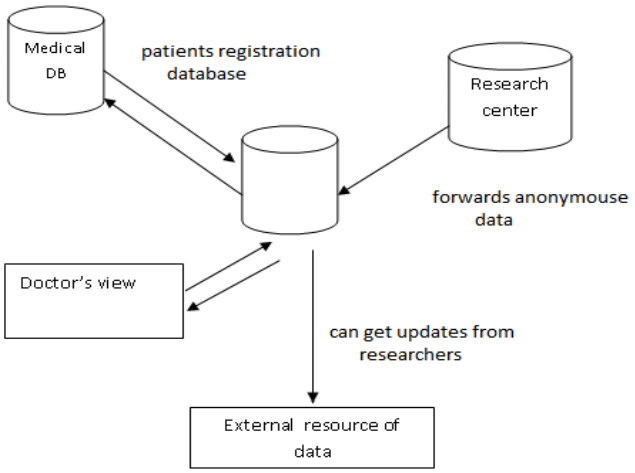

Fig -3: Architecture Diagram

RELATED WORK

The initial research work on this subject was conducted in [1]. Various protocols were discussed in this paper. However, the protocols discussed in this paper proved to be inefficient to meet the goal of this paper. The protocols mainly concentrated on suppression-based updates and gave little importance to generalization-based updates. As our paper gives equal importance to both the concepts, the ideas discussed in [1] were not of great use in this paper. In the present, paper we illustrate two efficient protocols , one that provides privacy to the database and the second that attains confidentiality. We also provide general security to the data by applying cryptographic algorithms. To maintain the confidentiality and privacy of the data, detailed research on various related papers were carried out. The analysis carried out on each paper is as follows- The primary research work deals with study of various anonymization algorithms for database anonymization. The concept of protecting and preserving databases through suppression of data and perturbation of data has been thoroughly studied in the field of statistical databases[2]. Similar theories have been proposed and supported in [3] where the idea of k-anonymization for databases has been studied and in [4] where complexity results concerning algorithms for k-anonymization has been developed. However, certain drawbacks in the usage of k-anonymization algorithms were explained in [5] where the major problem of this algorithm was considered to be maintenance of confidentiality of data during addressing of a particular tuple. The next level of research work was conducted in [6]. This paper studied the concept of Secure Multiparty Computation (SMC) technique.SMC deals with certain very important class of techniques that are used extensively in the field of cryptography. General methods to perform secure computation in data are now widely available[6]. However, these techniques lack the efficiency which is required for this implementation. These disadvantages has encouraged a thorough research in this area. As an outcome of the research, a particular protocol in the field of data management which makes accomplishment of the aim of this paper possible has been introduces in [8] and [9]. In these papers, the authors address the problems of efficiency without compromising on its confidentiality. The third level of research work deals with private information retrieval, which can be seen as a subpart of SMC. This field is useful mainly in the field of data management. Here, the main objective is to implement techniques for expressing efficient queries over a database without revealing the actual query to the database [7], [10]. However, the problem of addressing the tuple privately has not been solved through this method also. Finally, the ultimate research direction deals with methods for processing query for encrypted data[11], [12] and [13].These methods does not concern with k-anonymity technique as their basic purpose is to encrypt data, so that outsourcing of their management to external entities can be possible. Thus they fulfill the purpose of protecting data confidentiality from external entities which manage the data.

3 ALGORITHM

K-TECHNIQUE ALGORITHM

In this paper we are going to implement two protocols which will provide confidentiality along with privacy based on the k-technique algorithm. This algorithm contains two basic methods ,called Suppression based method and Generalization based method. Depending on these two main methods here two different protocols are used. and they have their different methodologies with restrictions and rules. One of them is private update for suppression-based anonymous and confidential database.

PROTOCOL FOR SUPPRESSION:

X codes tuple t ,into c((v1',....vn')), denoted as c(ti). Then ,she encrypts c(ti) with her private key and sends Ea(c(ti)) to Y.

Y individually codes each attribute value in m to get the tuple of coded values (c(v1),....c(vs)),encrypts each coding and Ea(c(ti)) with his key B and sends

(EB(c(v1)),.....,EB(c(vu))),and
EB(EA(c(ti))) to X.

Since E is a commutative encryption scheme
EB(EA(c(ti))) = EA(EB(c(ti))), x decrypts
EA(EB(c(ti))) to obtain EB(c(ti)).

Since the encrypted values sent by Y are ordered according to the ordering of the attributes in T .X knows which are, among the encrypted values sent by Y ,the one corresponding to the suppressed and non suppressed QI attributes . Thus, X computes

EB(c(v1)) *...*EB(c(vn))

Where (v1,…,vs) are the values of non-suppressed attributes contained in tuple t. As already mentioned, E is a product-homomorphic encryption scheme. Based also on definition of function c(.),this implies that previous Expression is equal to EB(c((v1,....,vn))).

X checks whether

EB(c((v1,....,vn))) = EB(c((v1',....,vn'))).

If true ,t can be inserted to table T. Otherwise , when inserted to T, t breaks k-anonymity.

PROTOCOL FOR GENERALIZATION:

1. X randomly chooses ad element of Tw.
2. X computes G=GetSpec(d)
3. X and Y collaboratively compute s=SSI(G,t)
4. if s=u then t's generalized form can be safely inserted to T.
5. Otherwise , X computes Tw=Tw-{d} and repeat the above procedure until either s=u or Tw= NULL .

The following example illustrates the execution of the protocol. Suppose X has the witness set Tw.

STEP 1: X encrypt the tuple T, and send it to Y.

STEP 2: Y can decrypt tuple T and then suppress the personal identifiers in the tuple.

STEP 3: After the suppression check the non-suppressed attributes in the tuple T and loaded tuples.

STEP 4: If any match found, insertion can be performed and send a status message "INSERTED".

STEP 5: If no match found, discard the tuple and send the status message "IGNORE".

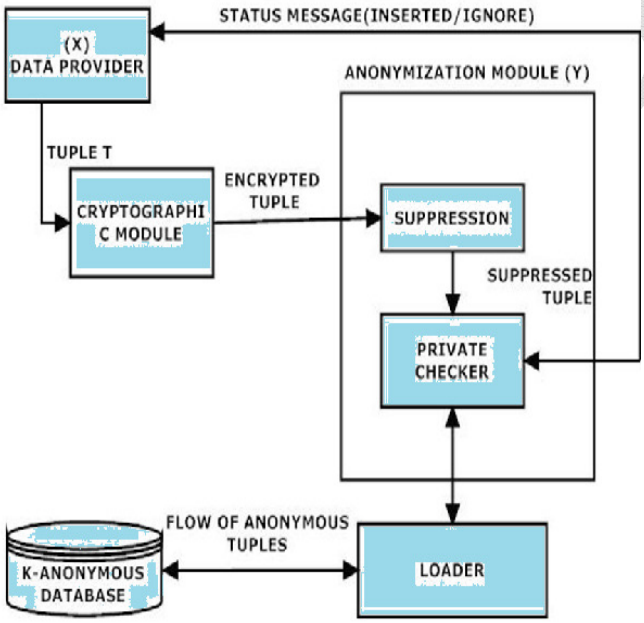

Fig -4: Proposed Methodology

CONCLUSION

In this paper two protocols have been presented by us for verifying whether a k anonymous database maintains its anonymity after inserting a new tuple to its database. For any database system to efficiently perform privacy preserving updates to anonymous table protocols for generalization and suppression are necessary but clearly not satisfactory. As previously mentioned in section 1 other essential activities are to be addressed. i) The definition of a mechanism for actually performing the update once k-anonymity has been verified. ii)The particularization of the actions t take in case of both the protocols yield adverse answer. iii) How to populate an empty table at the beginning. In the later stage we improvise on the solution produced in order to acknowledge these questions and which includes our overall organization for the updation of data. In the general advent we isolate the process of k-anonymity verification and the actual update differently into two phases. In the first phase database administrator ensuring both the protocol verifies for the k-anonymity database which is based on the result of the anonymity check. We assign to this as update execution. During every phase the user and the database system communicate through anonymous connection. The authorized are also authenticated anonymously through certain protocol. Regarding the original execution of the database updates. Once the DBA Alice in protocols has verified that the insertion of the users tuple is safe without compromising the k-anonymity, the second user Bob is made is made to send the DBA the non anonymous attributes values stored in the k-anonymous database as well. It has to be noted that the deployment of an anonymity system makes sure that the DBA cannot relate the sender of the tuple until the subject by whom the analogous insertion request has been made. In case the tuple fails the test of the protocols then the insertion of the tuple to k-anonymous database is not done by the DBA. At this point it is checked by the DBA whether such set of tuples refuse to as pending tuple set are k-anonymous . In the negative case the k-anonymization of the set of tuples failing the insertion is periodically checked again by method presented in. In the positive case the DBA proceeds to insert the k-anonymous tuples to the database. Many issues are to be concerned addressing the above methods to be efficient. Consider an example where and who is responsible for keeping the pending tuple set; how to inform and communicate with databases in order to initiate the protocol. This issues can be addressed more extensively in future.

ACKNOWLEDGMENT

The authors are sincerely grateful to Prof.B.S.Khade, our Project guide and mentor for her valuable guidance and encouragement. Also the authors are thankful to the Information Technology Department of JSPM's Bhivarabai Sawant Institute of Technology & Research (For Women) for their support in providing a good environment and facilities like books, internet and the other resources to complete this research.

REFERENCES

[1]. A. Trombetta and E. Bertino, "Private Updates to Anonymous Databases," Proc. Int'l Conf. Data Eng. (ICDE), 2006.

[2]. 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.

[3]. L. Sweeney, "k-Anonymity: A Model for Protecting

Privacy," Int'l J. Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 5,pp. 557-570, 2002.

[4]. 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.

[5]. S. Zhong, Z. Yang, and R.N. Wright, "Privacy-Enhancing k-Anonymization of Customer Data," Proc. ACM Symp. Principles of Database Systems (PODS), 2005.

[6]. O. Goldreich, Foundations of Cryptography: Basic Applications, vol. 2. Cambridge Univ. Press, 2004.

[7]. 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.

[8]. R. Agrawal, A. Evfimievski, and R. Srikant, "Information Sharing across Private Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2003.

[9]. M.J. Freedman, M. Naor, and B. Pinkas, "Efficient Private Matching and Set Intersection," Proc. Eurocrypt Conf., 2004.

[10]. U. Maurer, "The Role of Cryptography in Database Security," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2004.

[11]. D. Boneh, G. di Crescenzo, R. Ostrowsky, and G. Persiano, "Public Key Encryption with Keyword Search," Proc. Eurocrypt Conf., 2004.

[12]. 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.

[13]. D.X. Song, D. Wagner, and A. Perrig, "Practical Techniques for Searches on Encrypted Data," Proc. IEEE Symp. Security and Privacy, 2000.

[14]. M. Steiner, G. Tsudik, and M. Waidner, "Diffie-Hellman Key Distribution Extended to Group Communication," Proc. ACM Conf. Computer and Comm. Security, 1996.