International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 11, November 2014)

Privacy Preserving Distributed Data Mining with Anonymous ID Assignment

Chikkudu Chandrakanth
M.Tech Scholar(CSE)
Sri Indu College of Engg and Tech
Ibrahimpatan, Hyderabad, TS, India

Bheemari Santhoshkumar
M.Tech Scholar(CSE)
Sri Indu College of Engg and Tech
Ibrahimpatan, Hyderabad, TS, India

Tejavath Charan Singh
Assistant Professor, Dept of CSE
Sri Indu College of Engg and Tech
Ibrahimpatan, Hyderabad, TS, India

Abstract: **This paper builds an algorithm for sharing simple integer data on top of secure sum data mining operation using Newton's identities and Sturm's theorem. Algorithm for anonymous sharing of private data among parties is developed. This assignment is anonymous in that the identities received are unknown to the other members of the group. Resistance to collusion among other members is verified in an information theoretic sense when private communication channels are used. This assignment of serial numbers allows more complex data to be shared and has applications to other problems in privacy preserving data mining, collision avoidance in communications and distributed database access. The new algorithms are built on top of a secure sum data mining operation using Newton's identities and Sturm's theorem. An algorithm for distributed solution of certain polynomials over finite fields enhances the scalability of the algorithms.**

Key words: **Cloud, Website, information sharing, DBMS, ID, ODBC, ASP.NET**

.

1. INTRODUCTION

In recent years, we have observed an explosion of information shared among organizations in many realms ranging from business to government agencies. To facilitate efficient large-scale information sharing, many efforts have been devoted to reconcile data heterogeneity and provide interoperability across geographically distributed data sources. Meanwhile, peer autonomy and system coalition becomes a major tradeoff in designing such distributed information sharing systems. Most of the existing systems work on two extremes of the spectrum: in the query-answering model for on-demand information access, peers are fully autonomous but there is no system-wide coordination; so that participants create pair wise client-server connections for information sharing; in the traditional distributed database systems, all the participates lost autonomy and are managed by a unified DBMS. Unfortunately, neither of them is suitable for many newly emerged applications, such as information sharing for healthcare or law enforcement, in which organizations share information in a conservative and controlled manner, not only from business considerations but also due to legal reasons.

The popularity of internet as a communication medium whether for personal or business use depends in part on its support for anonymous communication. Businesses also have legitimate reasons to engage in anonymous communication and avoid the consequences of identity revelation. For example, to allow dissemination of summary data without revealing the identity of the entity the underlying data is associated with, or to protect whistle-blower's right to be anonymous and free from political or economic retributions .Cloud-based website Management tools provide capabilities for a server to anonymously capture the visitor's web actions. The problem of sharing privately held data so that the individuals who are the subjects of the data cannot be identified has been researched extensively. Researchers have also investigated the relevance of anonymity and/or privacy in various application domains: patient medical records, electronic voting, e-mail, social networking, etc. Another form of anonymity, as used in secure multiparty computation, allows multiple parties on a network to jointly carry out a global computation that depends on data from each party while the data held by each party remains unknown to the other parties., A secure computation function widely used in the literature is secure sum that allows parties to compute the sum of their individual inputs without disclosing the inputs to one another. This function is popular in data mining applications and also helps characterize the complexities of the secure multiparty computation.

This work deals with efficient algorithms for assigning identifiers (IDs) to the nodes of a network in such a way that the IDs are anonymous using a distributed computation with no central authority. Given N nodes, this assignment is essentially a permutation of the integers $\{1... N\}$ with each ID being known only to the node to which it is assigned. Our main algorithm is based on a method for anonymously sharing simple data and results in methods for efficient sharing of complex data. There are many applications that require dynamic unique IDs for network nodes. Such IDs can be used as part of schemes for Sharing/dividing communications bandwidth, data storage, and other resources anonymously and without conflict. The IDs are needed in sensor networks for security or for administrative tasks requiring reliability, such as configuration and monitoring of individual nodes. An application where IDs need to be anonymous is grid computing where one may seek services without divulging the identity of the service requestor. To differentiate anonymous ID assignment from anonymous communication, consider a situation where N parties wish to display their data collectively, but anonymously, in N slots on a third party site. The IDs can be used to assign the N slots to users, while anonymous communication can allow the parties to conceal their identities from the third party.

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 11, November 2014)

2. LITERATURE SURVEY

Providing k-anonymity in data mining: In recent years the data mining community has faced a new challenge. Having shown how effective its tools are in revealing the knowledge locked within huge databases, it is now required to develop methods that restrain the power of these tools to protect the privacy of individuals. This requirement arises from popular concern about the powers of large corporations and government agencies – concern which has been reflected in the actions of legislative bodies (e.g., the debate about and subsequent elimination of the Total Information Awareness project in the US). In an odd turn of events, the same corporations and government organizations which are the cause of concern are also among the main pursuers of such privacy- preserving methodologies. This is because of their pressing need to cooperate with each other on many data analytic tasks (e.g., for cooperative cyber-security systems, failure analysis in integrative products, detection of multilateral fraud schemes, and the like).The first approach toward privacy protection in data mining was to perturb the input (the data) before it is mined. Thus, it was claimed, the original data would remain secret, while the added noise would average out in the output. This approach has the benefit of simplicity. At the same time, it takes advantage of the statistical nature of data mining and directly protects the privacy of the data. The drawback of the perturbation approach is that it lacks a formal framework for proving how much privacy is guaranteed. This lack has been exacerbated by some recent evidence that for some data, and some kinds of noise, perturbation provides no privacy at all. Recent models for studying the privacy attainable through perturbation offer solutions to this problem in the context of statistical databases. At the same time, a second branch of privacy preserving data mining was developed, using cryptographic techniques. This branch became hugely popular for two main reasons: First, cryptography offers a well-defined model for privacy, which includes methodologies for proving and quantifying it. Second, there exists a vast toolset of cryptographic algorithms and constructs for implementing privacy-preserving data mining algorithms. However, recent work has pointed that cryptography does not protect the output of a computation. Instead, it prevents privacy leaks in the process of computation. Thus, it falls short of providing a complete answer to the problem of privacy preserving data mining.

Untraceable electronic mail, return address and digital pseudonyms: Cryptology is the science of secret communication. Cryptographic techniques have been providing secrecy of message content for thousands of years. Recently some new solutions to the "key distribution problem" (the problem of providing each communicant with a secret key) have been suggested, under the name of public key cryptography. Another cryptographic problem, "the traffic analysis problem" (the problem of keeping confidential who converses with whom, and when they converse), will become increasingly important with the growth of electronic mail. This paper presents a solution to the traffic analysis problem that is based on public key cryptography. Baran has solved the traffic analysis problem for networks, but requires each participant to trust a common authority. In contrast, systems based on the solution advanced here can be compromised only by subversion or conspiracy of all of a set of authorities. Ideally, each participant is an authority. The following two sections introduce the notation and assumptions. Then the basic concepts are introduced for some special cases involving a series of one or more authorities. The final section covers general purpose mail networks.

3. EXISTING SYSTEM

Existing and new algorithms for assigning anonymous IDs are examined with respect to trade-offs between communication and computational requirements. Also, suppose that access to the database is strictly controlled, because data are used 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; 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. This function is popular in data mining applications and also helps characterize the complexities of the secure multiparty computation. Our main algorithm is based on a method for anonymously sharing simple data and results in methods for efficient sharing of complex data. There are many applications that require dynamic unique IDs for network. Disadvantages of existing system the problem of sharing privately held data so that the individuals who are the subjects of the data cannot be identified has been researched extensively. The database with the tuple data does not be maintained confidentially. The existing systems another person to easily access database. The algorithms for mental poker are more complex and utilize cryptographic methods as players must, in general, be able to prove that they held the winning hand.

4. PROPOSED SYSTEM

An algorithm for anonymous sharing of private data among parties is developed. This technique is used iteratively to assign these nodes ID numbers ranging from 1 to N. This assignment is anonymous in that the identities received are unknown to the other members of the group. Resistance to collusion among other members is verified in an information theoretic sense when private communication channels are used. This assignment of serial numbers allows more complex data to be shared and has applications to other problems in privacy preserving data mining, collision avoidance in communications and distributed database access. The required computations are distributed without using a trusted central authority. Advantages of proposed system the anonymity of

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 11, November 2014)

DB is not affected by inserting the records. We provide security proofs and experimental results for both protocols.

5. SYSTEM ARCHITECTURE

3-tier application is a program which is organized into three major disjunctive tiers on layers. Here we can see that how these layers increase the reusability of codes. These layers are described below. Application layer or Business layer Business layer.
Property layer (Sub layer of business layer), Data layer

Advantages of three Tier Architecture: The main characteristic of a Host Architecture is that the application and databases reside on the same host computer and the user interacts with the host using an unfriendly and dump terminal. This architecture does not support distributed computing (the host applications are not able to connect a database of a strategically allied partner). Some managers found that developing a host application take too long and it is expensive. Consequently led these disadvantages to Client-Server architecture. Client-Server architecture is 2-Tier architecture because the client does not distinguish between Presentation layer and business layer. The increasing demands on GUI controls caused difficulty to manage the mixture of source code from GUI and Business Logic (Spaghetti Code). Further, Client Server Architecture does not support enough the Change Management. Let suppose that the government increases the Entertainment tax rate from 4% to 8 %, then in the Client-Server case, we have to send an update to each clients and they must update synchronously on a specific time otherwise we may store invalid or wrong information. The Client-Server Architecture is also a burden to network traffic and resources. Let us assume that about five hundred clients are working on a data server then we will have five hundred ODBC connections and several ruffian record sets, which must be transported from the server to the clients (because the Business layer is stayed in the client side). The fact that Client-Server does not have any caching facilities like in ASP.NET caused additional traffic in the network. Normally, a server has a better hardware than client therefore it is able compute algorithms faster than a client, so this fact is also an additional pro argument for the 3.Tier Architecture. This categorization of the application makes the function more reusable easily and it becomes too easy to find the functions which have been written previously. If programmer wants to make further update in the application then he easily can understand the previous written code and can update easily.

Application layer or Presentation layer: Application layer is the form which provides the user interface to either programmer of end user. Programmer uses this layer for designing purpose and to get or set the data back and forth.

Business layer: This layer is a class which we use to write the function which works as a mediator to transfer the data from Application or presentation layer data layer. In the three tier architecture we never let the data access layer to interact with the presentation layer.

Property Layer: This layer is also a class where we declare the variable corresponding to the fields of the database which can be required for the application and make the properties so that we can get or set the data using these properties into the variables. These properties are public so that we can access its values.

Data Access Layer: This layer is also a class which we use to get or set the data to the database back and forth. This layer only interacts with the database. We write the database queries or use stored procedures to access the data from the database or to perform any operation to the database.

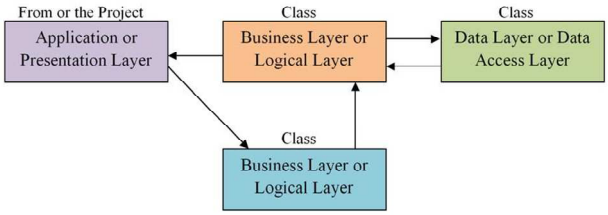

Figure 1: 3-Tier Architecture summary

- Application layer is the form where we design using the controls like textbox, labels, command buttons etc.
- Business layer is the class where we write the functions which get the data from the application layer and passes through the data access layer.
- Data layer is also the class which gets the data from the business layer and sends it to the database or gets the data from the database and sends it to the business layer.
- Property layer is the sub layers of the business layer in which we make the properties to sent or get the values from the application layer. These properties help to sustain the value in a object so that we can get these values till the object destroy.

Data flow from application layer to data layer: You can download sample three tier project, used for this tutorial. Here we are passing the code of the student to the business layer and on the behalf of that getting the data from the database which is being displayed on the application layer.

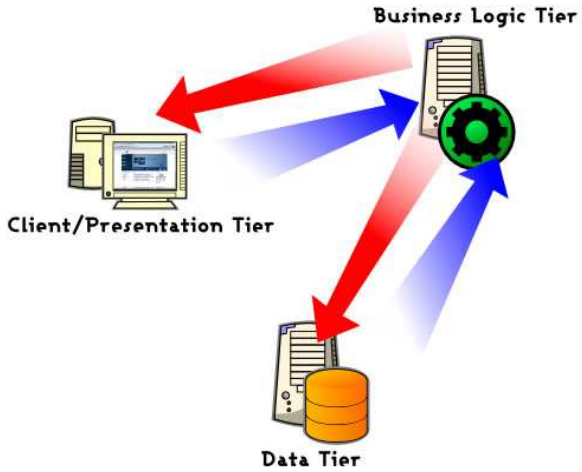

Figure 2: Dataflow from Application to Data layer

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 11, November 2014)

Data sets: The data fetched from a data store is placed in a Dataset. A Dataset can contain multiple tables, and even the relationships between them. It thus reflects the data store in the disconnected or offline mode and enables a large amount of data to be passed across the applications in one go. This data is stored in the memory of the system. The data in the Dataset can be passed between applications in XML format.

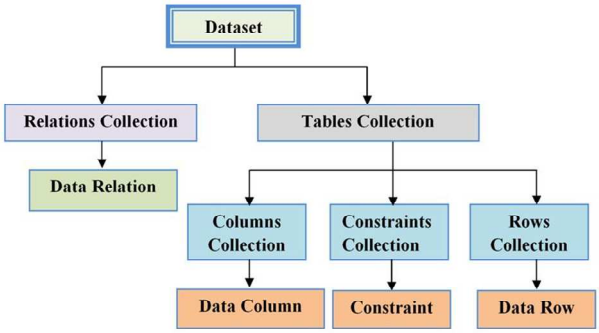

Fig.3. Datasets in ASP.NET

A Dataset however need not always be linked to a data store. It can be created programmatically and contain any kind of data. In spite of its similarities to a Record set, a Dataset is however not a Record set. Record set contains (hierarchical) collection of records of a table. A Dataset can be considered a super-Record set. It contains multiple tables, views and the relationships between the tables. The Dataset object is dynamically created by the ADO+ runtime using the XML schema for a specific platform. Thus, data can be safely exchanged between different platforms. Only the data is exchanged between the platforms in XML format and the Dataset used to represent the data is created on the receiving platform in native format.

6. SCREENSHOTS

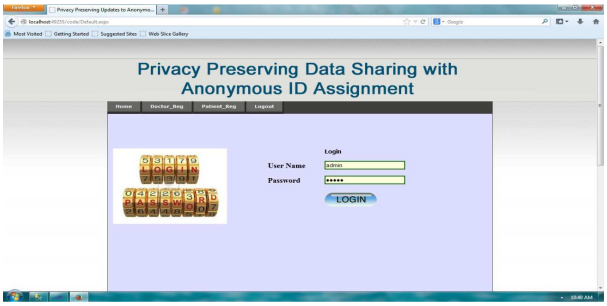

Figure 4: Admin login page

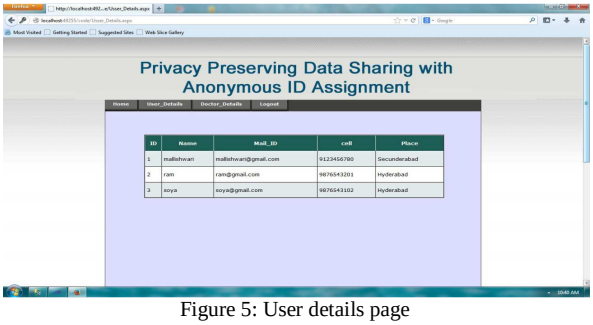

Figure 5: User details page

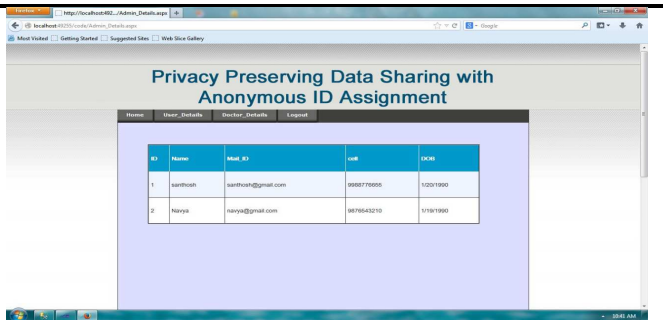

Figure 6: Doctor details page

Figure 7: Doctor Registration page

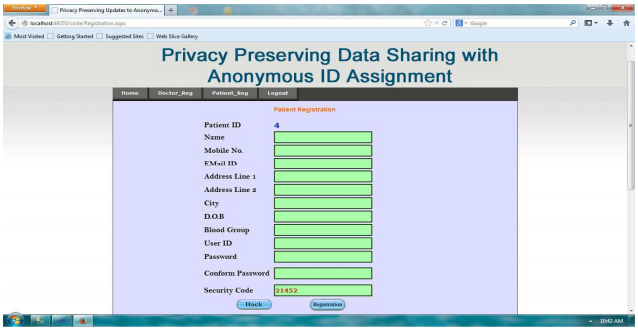

Figure 8: Patient registration page

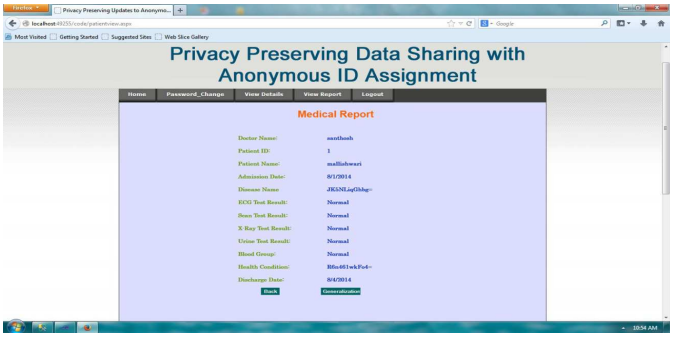

Figure 9: User details encrypted medical report page

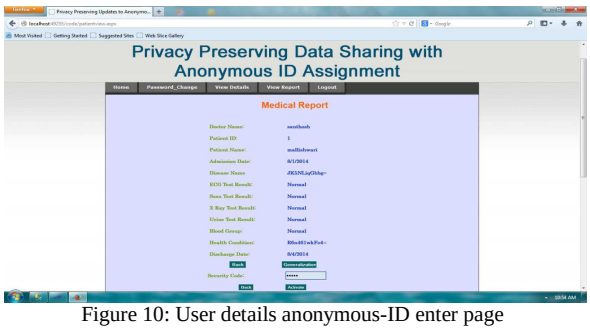

Figure 10: User details anonymous-ID enter page

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 11, November 2014)

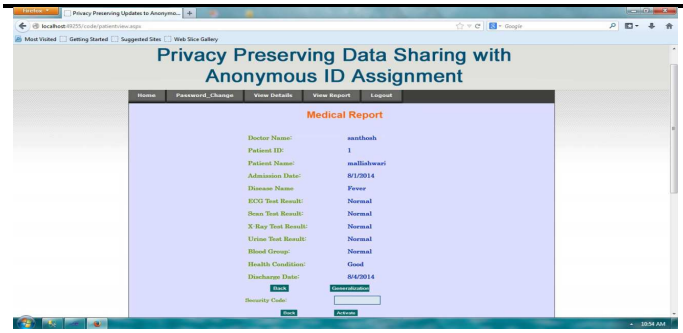

Figure 11: User details medical report page

7. CONCLUSION

Each algorithm compared in Section VI can be reasonably implemented and each has its advantages. Our use of the Newton identities greatly decreases communication overhead. This can enable the use of a larger number of "slots" with a consequent reduction in the number of rounds required. The solution of a polynomial can be avoided at some expense by using Sturm's theorem. The development of a result similar to the Sturm's method over a finite field is an enticing possibility. With private communication channels, our algorithms are secure in an information theoretic sense. Apparently, this property is very fragile. The very similar problem of mental poker was shown to have no such solution [22] with two players and three cards. The argument of [22] can easily be extended to, e.g., two sets each of colluding players with a deck of cards rather than our deck of cards. In contrast to bounds on completion time developed in previous works, our formulae give the expected completion time exactly. We conjecture the asymptotic formula of Corollary 9, based on computational experience, to be a true upper bound. All of the no cryptographic algorithms have been extensively simulated, and we can say that the present work does offer a basis upon which implementations can be constructed. The communications requirements of the algorithms depend heavily on the underlying implementation of the chosen secure sum algorithm. In some cases, merging the two layers could result in reduced overhead.

REFERENCES

[1] Beginning ASP.NET 4: in C# and VB by Imar Spaanjaars.
[2] Programming ASP.NET 3.5 by Jesse Liberty, Dan Maharry, Dan Hurwitz.
[3] Beginning ASP.NET 3.5 in C# 2008: From Novice to Professional, Second Edition by Matthew MacDonald.
[4] Sarbanes–Oxley Act of 2002, Title 29, Code of Federal Regulations, Part 1980, 2003.
[5] F. Baiardi, A. Falleni, R. Granchi, F. Martinelli, M. Petrocchi, and A. Vaccarelli, "Seas, a secure e-voting protocol: Design and implementation," Comput. Security, vol. 24, no. 8, pp. 642–652, Nov. 2005.
[6] D. Chaum, "Untraceable electronic mail, return address and digital pseudonyms," Commun. ACM, vol. 24, no. 2, pp. 84–88, Feb. 1981.
[7] Q. Xie and U. Hengartner, "Privacy-preserving matchmaking for mobile social networking secure against malicious users," in Proc. 9th Ann. IEEE Conf. Privacy, Security and Trust, Jul. 2011, pp. 252–259.
[8] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game," in Proc. 19th Ann. ACM Conf. Theory of Computing, Jan. 1987, pp. 218–229, ACM Press.