International Journal of Innovative Research in Computer Science and Engineering (IJIRCSE)
www.ioirp.com ISSN: 2394-6364, Volume – 1, Issue – 6. May 2015

Efficient and Provably Secure Aggregation of Encrypted Data Using Without Secure Channel

S.Vanmathimanju[1], P.Sumathi[2], S.Kalaiyarasi[3], A.Joseph selva kumar[4]
[13]PG Scholar, Department of CSE, Anna University, Idhaya Engineering College for Women, India
[24]Assistant Professor, Department of CSE, Anna University, Idhaya Engineering College for Women, India
tamilvanmathi@gmail.com, skalaics@gmail.com, ajosephsk@gmail.com

Abstract—Much research has been conducted to securely outsource multiple parties' data aggregation to an un trusted aggregator without disclosing each individual's privately owned data, or to enable multiple parties to jointly aggregate their data while preserving privacy. However, those works either require secure pair-wise communication channels or suffer from high complexity. In this paper, to consider how an external aggregator or multiple parties can learn some algebraic statistics (e.g., sum, product) over participants' privately owned data while preserving the data privacy. Assume all channels are subject to eavesdropping attacks, and all the communications throughout the aggregation are open to others. In first propose several protocols that successfully guarantee data privacy under semi-honest model, and then present advanced protocols which tolerate up to k passive adversaries who do not try to tamper the computation. Under this weak assumption, to limit both the communication and computation complexity of each participant to a small constant. At the end, to present applications which solve several interesting problems via the protocols.

Keywords— Privacy, data aggregation, secure channels, SMC, homomorphic.

I. INTRODUCTION

The Privacy-preserving data aggregation problem has long been a hot research issue in the field of applied cryptography. In numerous real life applications such as crowd sourcing or mobile cloud computing, individuals need to provide their sensitive data (location-related or personal-information-related) to receive specific services from the entire system (e.g., location based services or mobile based social networking services). There are usually two different models in this problem: 1) an external aggregator collects the data and wants to conduct an aggregation function on participants' data (e.g., crowd sourcing); 2) participants themselves are willing to jointly compute a specific aggregation function whose input data is co-provided by themselves (e.g., social networking services). However, the individual's data should be kept secret, and the aggregator or other participants are not supposed to learn any useful information about it. Secure Multi-party Computation (SMC), Homomorphic Encryption (HE) and other cryptographic methodologies can be partially or fully exploited to solve this problem, but they are subject to some restrictions in this problem.

Secure Multi-party Computation (SMC) is enables n parties who want to jointly and privately compute a function. Secure pair-wise communication channels. The main contributions of this paper are:

- Formulation of a model without secure channel or trusted centre Different from many other models in privacy preserving data aggregation problem, model does not require a secure communication channel nor a trusted central key issuer.
- Efficient protocol in linear time: The total communication and computation complexity of the work is proportional to the number of participants n, while the complexities of many similar works are proportional to n
- Secure sum and product calculation: To generalize the privacy-preserving data aggregation to multivariate sum and product calculation whose inputs are jointly provided by multiple parties. That the scheme enables multiple parties to securely compute
- Tolerate up to k collusive adversaries: The protocol is robust against up to k colluding passive adversaries who do not try to tamper the computation.

II. PROPOSED SYSTEM

A. Problem Definition and Threat Model

Assume that there are n participants and each participant has a privately known data from . The privacy-preserving data aggregation problem is to compute sum or product jointly by an aggregator while preserving the data privacy. That is, the objective of the aggregator or the participants is to compute the following polynomial without knowing any individual

B. Achieving sum & product under secure channel

Introducing aggregation scheme without secure communication channel, first describe the basic idea of randomized secure sum calculation under secured communication channel (It can be trivially converted to secure product calculation) The receiving participant adds all its received segments and transmits its result to the next participant in the ring. This process is repeated until all the segments of all the participants are added and the sum is announced by the aggregator.

C. Sum and product calculation without secure channel

Two novel calculation protocols for each model which calculate sum and product while preserving each participant's data privacy against semi-honest adversaries who follow the protocol specification and do not collude with anyone. To defend against passive adversaries who may adaptively choose their secret parameters based on others' public parameters and collude with each other.

D. Product Protocol - One Aggregator Model

The product calculation in the one aggregator model is similar to the protocol above, except that the aggregator and each participant will send the cipher text to the aggregator, instead of broadcasting to all participants. However, since the communication channel is insecure, this is essentially the broadcast from the adversary's perspective. The last difference is that the aggregator will not announce its random number to any other participant.

E. Security of Product Protocol

Since the communication channel is insecure, any adversary has same view in both Participants Only Model and One Aggregator Model. Then analyze the security of Participants Only Model in this section.

F. One Aggregator Model

It is easy to see that the computation complexities of Encrypt and Product of the product protocol and Encrypt is executed for many times by each participant and Product is executed for many times by the aggregator in the advanced scheme and respectively, and they are executed for only once in the scheme.

G. Participants Only Model

Participants Only Model, participants broadcast cipher text to others, and calculates the products and sums . To compare the performance of the protocol with other existing multi party computation system are implemented. They implemented the BMR protocols are required the constant number of communication rounds .The function being computed. Their system provides a platform for general secure multi-party computation (SMC), where one can program their secure computation with Secure Function Definition Language (SFDL).The programs wrote in SFDL enable multiple parties to jointly evaluate an arbitrary sized boolean circuit.

III. MODULE DESCRIPTION

A. Secure Multi-party Computation

Secure multi-party computation (also known as secure computation or multi-party computation (MPC)) is a subfield of cryptography. The goal of this field is to create methods that enable parties to jointly compute a function over their inputs, while at the same time keeping these inputs private. The security of a two-party computation protocol is usually defined through a comparison with an idealized scenario that is secure by definition. The idealized scenario involves a trusted party that collects the input of the two parties over secure channels and returns the result if none of the parties chooses to abort. The cryptographic two-party computation protocol is secure, if it behaves no worse than this ideal protocol, but without the additional trust assumptions.

This is usually modeled using a simulator. The task of the simulator is to act as a wrapper around the idealized protocol to make it appear like the cryptographic protocol .

B. Privacy Data Aggregation

An external aggregator collects the data and wants to conduct an aggregation function on participants' data (e.g., crowd sourcing). Participants themselves are willing to jointly compute a specific aggregation function whose input data is co-provided by them. Privacy preserving data aggregation or in general secure multiparty computation.

C. Data sharing

In this module, the user can only access the data file with the encrypted key if the user has the privilege to access the file. For the user level, all the privileges are given by the Domain authority and the Data users are controlled by the Domain Authority only. Users may try to access data files either within or outside the scope of their access privileges, so malicious users may collude with each other to get sensitive files beyond their privileges.

D. Cluster-Based Private Data Aggregation

In this module, In CPDA, sensor nodes form clusters randomly and collectively compute the aggregate result within each cluster. In the improved SMART, each node segments its data into n slices and distributes n1 slices to nearest nodes via secure channel. However, they only support additions, and since each data is segmented, communication overhead per node is linear to the number of slices n. A cluster-based private data aggregation (CPDA) scheme in which the sensor nodes are randomly distributed into clusters .The cluster leaders are responsible for directly aggregating data from the cluster members, with the communication secured by a shared key between a pair of communicating nodes. The aggregate function to compute the desired aggregate value in a cluster.

E. Data access

The cloud service provider manages a cloud to provide data storage service. Data owners encrypt their data files and store them in the cloud for sharing with data consumers. To access the shared data files, data consumers download encrypted data files of their interest from the cloud and then decrypt them. In the hierarchical structure of the system users given each party is associated with a public key and a private key, with the latter being kept secretly by the party. The trusted authority acts as the root of trust and authorizes the top-level domain authorities. When the sharer wants to access the file, he sends a request to the cloud server. The cloud determines the validity of the sharer by checking if it has a re-encryption key to the sharer. With the re-encryption key is existed, the cloud server can run the RK Gen algorithm and achieve the re-encryption cipher text.

IV. STATISTICS CALCULATION

Privacy-preserving statistics calculation is desired in various applications of protocols can be directly used to express various statistical values. Directly computed by the sum protocol computed with privacy preservation. Challenge query distribution

A. General Boolean Formula Evaluation

A boolean formula consists of True or False variables and logical operators (AND, OR, XOR etc.). It is important to securely evaluation a boolean formula in many problems (Multi-party Millionaire problem, Anonymous voting problem), and how to achieve a general Boolean formula evaluation without disclosing individuals' input values multi-party computation and the second homomorphic encryption system is for general homomorphic encryption. And also provide a much higher level of security than to achieve differential privacy, however, the comparison above does show the high speed of the system while the security level is still acceptable in real life applications, and this is one of the main contributions of this paper.

B. Veto Protocol

A privacy-preserving veto protocol requires that only the final outcome (whether there exists any veto) is published without disclosing any individual vote. It can be easily implemented by employing the above Boolean formula evaluation as a building block.

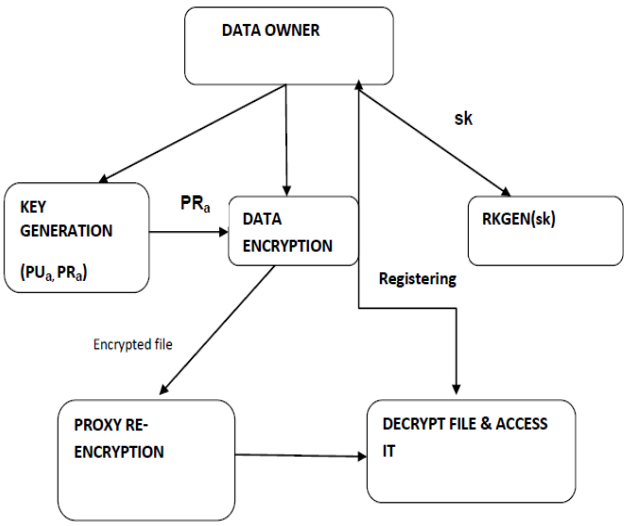
C. Millionaire Problem

The traditional millionaire problem requires to tell who is the richer between two millionaire while neither party knows the exact amount of money the other party possesses. This problem seems impossible to solve using the protocol since to require at least 3 participants in the participants only model, and to solve this with a
simple trick.

D. Evaluation by Implementation

To conduct extensive evaluations of the protocols. The simulation result shows that the computation complexity of the protocol is indeed linear to the number of participants. To simulate and measure the computation overhead, to used GMP library to implement large number operations in the protocol and, to measured the total overhead
of the product protocol and sum protocol to measured the total computation time spent in calculating the final result of n data in the evaluation by implementation

E. Architecture diagram of proposed system



F. Sub linear Solution

To improve the complexity by using the Veto Protocol above as a building block. One guesses a number x as the maximum value and asks every participant whether someone has a larger (smaller) value, any participant having a larger (smaller) value vetoes to the request. No one vetoes at x does not mean x is the maximum (minimum). According to the actual application, either participants only model or one aggregator model can be used in the secure polynomial evaluation protocol. To use the one aggregator model and let participants know which number is the maximum (minimum) value, we have to use the Linear Solution. In the Linear Solution, the aggregator can send dummy requests after finding the maximum (minimum) value to hide the value, but in the Sub linear Solution, participants immediately know the maximum (minimum)value after the ⌈log v⌉ rounds of the requests. Unlike the Linear Solution, no further dummy requests are possible since the search space is already narrowed down such that only one candidate is left.

V. CONCLUSION

In this paper, successfully achieve privacy-preserving sum and product calculation protocols without secure communication channels or trusted key issuers. To allow up to (adjustable parameter) collusive participants who will not tamper the computation but try to manipulate their parameters to infer others' private values. To analyzed the security of the protocols and showed that the protocols are secure if the CDH problem is assumed to be intractable, and also showed with implementation that the protocols are efficient to be applicable in real life. At the end, to propose numerous applications that are achieved from the protocols

A. Future Work

Future works is to design privacy preserving data releasing protocols such that general function of data can be evaluated correctly while preserving individuals' data privacy.

References

[1] C. C. Aggarwal and S. Y. Philip, A general survey of privacy preserving data mining models and algorithms. Springer, 2008.
[2] D. Beaver, S. Micali, and P. Rogaway, "The round complexity of secure protocols," in STOC. ACM, 1990.
[3] A. Ben-David, N. Nisan, and B. Pinkas, "Fairplaymp: a system for secure multi-party computation," in CCS. ACM, 2008.
[4] D. Boneh, "The decision diffie-hellman problem," Algorithmic Number Theory, 1998.
[5] C. Castelluccia, A. Chan, E. Mykletun, and G. Tsudik, "Efficient and provably secure aggregation of encrypted data in wireless sensor networks," Transactions on Sensor Networks (TOSN), 2009.
[6] C. Castelluccia, E. Mykletun, and G. Tsudik, "Efficient aggregation of encrypted data in wireless sensor networks," in MobiQuitous. IEEE, 2005.
[7] T.-H. H. Chan, E. Shi, and D. Song, "Privacy-preserving stream aggregation with fault tolerance," in Financial Cryptography and Data Security (FC). Springer, 2012.
[8] X. Chen, X. Wu, X.-Y. Li, Y. He, and Y. Liu, "Privacy-preserving high-quality map generation with participatory sensing," in INFOCOM. IEEE, 2014.
[9] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Zhu, "Tools for privacy preserving distributed data mining," SIGKDD
[10] Explorations Newsletter, 2002.
[11] C. Dwork, "Differential privacy," in Automata, languages and programming. Springer, 2006.
[12] C. Dwork and J. Lei, "Differential privacy and robust statistics," in STOC. ACM, 2009.
[13] B. Edelman, M. Ostrovsky, and M. Schwarz, "Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords," National Bureau of Economic Research, Tech. Rep., 2005.
[14] B. J. Falkowski, "A note on the polynomial form of Boolean functions and related topics," Transactions on Computers, 1999.
[15] N. Fazio, R. Gennaro, I. M. Perera, and W. E. Skeith III, "Hardcore predicates for a diffie-hellman problem over finite fields." 2013.
[16] J. Feigenbaum and M. Merritt, Distributed Computing and Cryptography: Proceedings of a Dimacs Workshop October 4-6, 1989. AMS
[17] Bookstore, 1991.
[18] M. Fischlin, "A cost-effective pay-per-multiplication comparison method for millionaires," in Topics in CryptologyCT-RSA 2001.
[19] A. Friedman and A. Schuster, "Data mining with differential privacy," in SIGKDD. ACM, 2010.
[20] B. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," Computing Surveys (CSUR), 2010.
[21] C. Gentry, "Fully homomorphic encryption using ideal lattices," in STOC. ACM, 2009.
[22] C. Gentry and S. Halevi, "Implementing gentrys fullyhomomorphic encryption scheme," Advances in Cryptology–EUROCRYPT, 2011.