IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 9, Issue 1 (Jan. - Feb. 2013), PP 49-52
www.iosrjournals.org

Distributed R_K- Secure Sum Protocol for Privacy Preserving

Jyotirmayee Rautaray[1], Raghvendra Kumar[1]
[1]*School of Computer Engineering, KIIT University, Odisha, India*

Abstract **:** *Secure multi party computation allows several parties to compute some function of their inputs without disclosing the actual input to one another. Secure sum computation is an easily understood example and the component of the various secure multi party computation solutions. Secure sum computation allows parties to compute the sum of their individual inputs without disclosing the inputs to one another. In this paper, we propose a protocol with more security, when a group of the computing parties want to know the data of some other party.*

Keywords:*Distributed Database Partition, Multi Party Computation, Privacy Preserving, Secure Sum Protocol.*

I. INTRODUCTION

The huge growth of the Internet and its easy access by common man created opportunities for joint computations by multiple parties. All the participating parties for the sake of their mutual benefit want to compute the common function of their inputs but at the same time they are worried about the privacy of their data. This subject of the information security is called secure multi party computation. This subject has two goals; one is the privacy of the individual data inputs and another is the correctness of the result. Mainly two models exist in the literature for the analysis of the secure multi party computation problems. Ideal model of the secure multi party computation uses a Trusted Third Party apart from the participating parties. Parties supply their inputs to the Trusted Third Party. Computation of the function is done by the Trusted Third Party and then the result is sent to all the parties. In this paradigm the trustworthiness of the Trusted Third Party is critically important because if the Trusted Third Party turns corrupt, it can supply the private inputs of one party to others. But it is extensively used model of the secure multi party computation due to its easy implementation and the protocols available which prevent the Trusted Third Party to act maliciously. Real model of the secure multi party computation does not use any Trusted Third Party but the parties themselves agree on some protocol for the computation. The parties behavior in the secure multi party computation is important to consider. An honest party follows the protocol and respects the privacy of other parties. A semi honest party follows the protocol but also tries to learn other information than the result. The corrupt party neither follows the protocol nor respects the privacy of other parties. Different protocols are needed for different secure multi party computation models and the behavior of the party. Solutions are available for secure multi party computation problems using cryptographic techniques, randomization techniques and anonymization methods. The subject of secure multi party computation has been evolved from two party comparison problems [1] to multiparty image template matching problems. Many specific secure multi party computation problems have been defined and analyzed by researchers like Private Information Retrieval, Selective Function Evaluation, Privacy-Preserving Database Query, Privacy-Preserving Geometric Computation, Privacy- Preserving Statistical Analysis, Privacy-Preserving Intrusion Detection and Privacy-Preserving Cooperative Scientific Computation. Based on these general secure multi party computation problems many real life applications have been emerged like Privacy- Preserving Electronic Voting, Privacy-Preserving Bidding and Auctions, Privacy-Preserving Social Network Analysis, Privacy-Preserving Signature and Face Detection, Etc. Secure sum computation problem of secure multi party computation can be defined as: How multiple parties can compute the sum of their input values without disclosing actual values to one another. Secure sum can work as the tool for the secure multi party computation solutions in the privacy preserving distributed data mining problems [2]. We proposed a secure sum protocol using random numbers [2]. In this paper, we proposed a novel changing neighbors approach Distributed RK secure sum protocol for achieving more security in case a group of the parties collude to know the private data of some other party.

1.1 Distributed Database Partition: - Database is divided into three main portioning horizontal partition, vertical partition and hybrid partition.

1.1.1 Horizontal Partitioning: - Horizontal partitioning divides the whole database into the number of small database according to the row splitting. So that the execution of query will be very fast as well as we will be able to provide more privacy to the portioned database.

1.1.2 Vertical Partitioning: - Vertical portioning divides the whole database into a number of small databases according to the column. So that portioned database does not contain any of duplicate data. There are mainly two types of vertical database normalized and row splitting.

1.1.3 Hybrid Partitioning: - hybrid partitioning first divide the database into horizontal portioning and then vertical or first vertical and then horizontal portioning it's depend upon the user requirements.

II. BACKGROUND WORK

The secure multi party computation started when the two millionaire's party wants the result of each other without disclosing the individual result. The concept of two parties [1] will be extended [6] after they are using the secure sum and secure multi party computation. It is one of the security protocol used in classification for preserving privacy of censored data. To provide extra security and preserve individual data, this protocol was modified in [2].

Let us consider a scenario where more than two parties (say n) want to compute and use classification technique in a secured way. So according to the protocol [2] [14] [15], first party adds a random number to its value and sends the result to second party. Second party adds its result to the imported result and sends to the third party. This process continues until the last party (n) receives the result from the previous party (n-1). This party adds its result and sends the output to next party. Secure multi party computation has two main goals one is provide the security to individual data and another is correctness of result. As well as secure multi party computation contain two main models one is real model and another is ideal model in real model there is no any trusted party but in ideal model there is trusted party is present. The whole secure multi party computation divides into three parts according to their inputs. The first one is convert the inputs into secure multi party computation after the computation. And second one is converting the inputs into homogeneous secure computation and another is converting the inputs into heterogeneous secure computations [8] [9] [11]. Fig1 shows the snapshot of secure multi party computation.

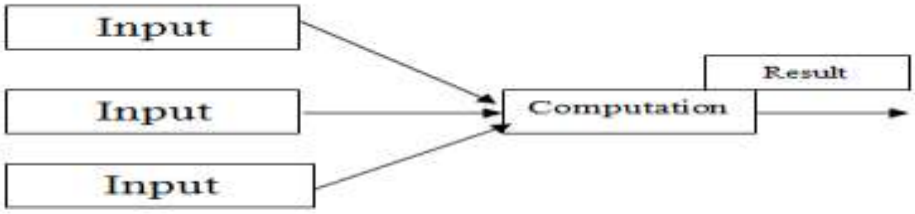

Fig1: - Snapshot of Secure Multi Party Computation

III. PROPOSED WORK

Informal Description- This Distributed RK secure sum protocol used bus architecture. All the parties (P1, P2……Pn) are connected to the Bus network. Select P1 as a protocol initiator and Pn are end party. First that P1 will calculated the result and send to the next party after adding its random variable or random number that presents in the network and this process will continue till the last party Pn. For first round the parties are arranged in P1, P2…… Pn in a Bus network. And the computation start from the first party and till Pn and Pn will store the result. In second round the P2 will exchange it's to the next party in the network and till process will continue till Pn. So that if the number of parties is N then the total number of rounds is N-1, So that the two nighoubour party will never know result of each other. And finally the result will allowanced by the P2. Snapshot of five parties is shown in figure 2 shows Distributed RK secure sum protocol. The protocol provides privacy against two colluding neighbors [9] [10]. Its analysis shows that when more than two parties join together, they can know the data of some party. The last party can be attacked by more than two parties that unkindly cooperate to know secret data of the last party or ending protocol. But for that also a specific combination of the parties must join against the last party. Any party who moves its position cannot be attacked by any group of the parties. Figure 2 shows that after exchanging of second party position to the third party. And figure 2 shows that after exchanging the position of party P2 to P4, figure 2 is shows after the result Distributed RK secure sum protocol of last round in this P2 is a last party and P2 will allowance the result.

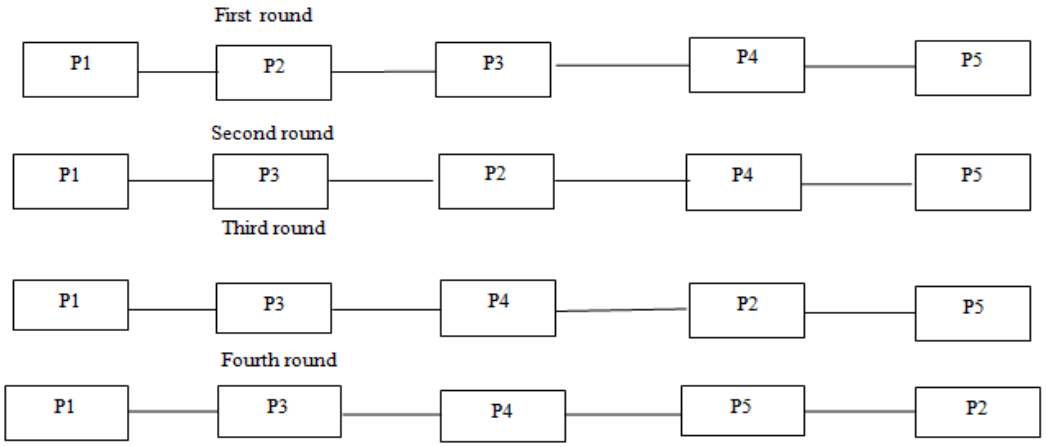

Fig 2: Snapshot of Movement of Five Parties

Formal Description ALGORITHM (Distributed RK secure sum Protocol)-
STEP1:- Assume the number of parties is n, P1, P2, P3……Pn (n≥3).
STEP2:- Assume these parties have their secrete random number R1, R2, R3……..Rn.
STEP3:- Each parties (P1, P2……Pn) have their data D1, D2……..Dn.
STEP4:- Arrange the parties in a bus structure (P1, P2…….Pn) and select P1 as a protocol initiator.
STEP5:- Assume that RC=n and Pi (RC is a round counter and Pi is partial support).
Partial support will calculate using the following formula

$$P_i = P_{(i-1)} + RC$$

STEP6:-While RC! =0
Begin
Begin
For 1 to n do
Begin
Starting from P1 each party will compute their partial support and send to the next party in the bus
End
P2 exchange its position to the next party present in the bus till Pn.
End
RC=RC-1
End
STEP7:- Party P2 will announce the result after calculating the from all the parties.
STEP8:- End of algorithm

IV. Comparison Between The Existing One And Proposed One

I Existing one is using the Ring topology [9] [10] [11] but the Distributed RK secure sum Protocol is using the Bus topology so that it's easy to design the Bus architecture as compared to Ring architecture because in bus the all parties are connected in a serial manner.
II In Existing one the segmentation of data is occurred for each party but in the Distributed RK secure sum Protocol the segmentation of data is not necessary because if segmentation of data is done then required the highest privacy as compare to transfer block of data at once.
III In block of data not required as much of privacy and it's more secure as compare of segments of data.
IV Number of round in Distributed RK secure sum Protocol is lesser as compare to the existing one because in a single round that not required to n number of data segment round again.
V The complexity of the Distributed RK secure sum Protocol is also reduce as compare to the existing one because the segmentation of data is not required in the proposed one.

V. Performance Analysis

In this Distributed RK secure sum Protocol the number of parties is N and the number of rounds is (N-1) because each party will exchange its position to the next party presents in the bus network. The only one limitation of this bus topology is that each rounds exchange its position. The computation and communication complexity both come to N. thus we can write the communication complexity C (n) and computation complexity S (n) are shown in figure 2.

C (N) =N
S (N) =N

The communication and computation complexity both come in order of N

VI. Conclusion And Future Work

Secure sum computation is an important component of secure multi party computation. The secure sum protocols are needed for secure sum computation with lower probability of data leakage. In this paper Distributed RK secure sum Protocol provides zero probability of data leakage by two or more colluding parties which want to know the data of some party. In this Distributed RK secure sum Protocol, the data block of each party is broken into certain number of segments and computation is performed over these segments. The parties are allowed to change their position in the ring. This ensures that a party cannot have same neighbors for all the rounds of the computation. Thus two or more colluding parties cannot learn the secret data of some other party. This is an improvement over previous protocols which ensure safety for two colluding neighbors only. Further efforts can be done to design and analyze the protocol for malicious parties who neither follow the protocol nor honor the privacy of the parties. Protocols can be designed to make the data secure in case majority of the parties are semi honest.

References

[1] A.C.Yao, "protocol for secure computations," in *proceedings of the 23rd annual IEEE symposium on foundation of computer science, 1987*, pp. 160-164.

[2] C. Clifton, M. Kantarcioglu, J.Vaidya, X. Lin, and M. Y. Zhu, "Tools for Privacy-Preserving Distributed Data Mining," *J.SIGKDD Explorations, Newsletter, vol.4, no.2*, ACM Press, 2002, pp.28-34.

[3] R. Sheikh, B. Kumar and D. K. Mishra, "Changing Neighbors k- Secure Sum Protocol for Secure Multi-party Computation," Accepted for publication in *the International Journal of Computer Science and Information Security, USA, Vol.7 No.1*, 2010, pp. 239-243.

[4] R. Sheikh, B. Kumar and D. K. Mishra, "Privacy-Preserving k- Secure Sum Protocol," in the International *Journal of Computer Science and Information Security, USA, Vol. 6 No.2*, 2009, pp. 184-188.

[5] R. Sheikh, B. Kumar and D. K. Mishra, "A Distributed k-Secure Sum Protocol for Secure Multi-party Computation," *submitted to a journal*, 2009.

[6] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game," in *STOC '87: Proceedings of the nineteenth annual ACM conference on Theory of computing, New York, NY, US* : ACM, 1987, pp. 218-229.

[7] B.Chor and N.Gilbao. "Computationally Private Information Retrieval (Extended Abstract)," In *proceedings of 29th annual ACM Symposium on Theory of Computing, El Paso, TX USA*, May 1997.

[8] B. Chor, E. Kushilevitz, O. Goldreich,and M. Sudan, "Private Information Retrieval ," In *proceedings of the 36th Annual IEEE Symposium on Foundations of Computer Science*, Milwaukee WI, 1995,pp. 41-50.

[9] Y. Lindell and b. Pinkas, "Privacy preserving data mining," in *advances in cryptography-Crypto2000, lecture notes in computer science*, Vol. 1880.

[10] R. Agrawal and R. Srikant. "Privacy-Preserving Data Mining," In *proceedings of the 2000 ACM SIGMOD on management of data, Dallas, TX USA, 2000*, pp. 439-450.

[11] M. J. Atallah and W. Du. "Secure Multiparty Computational Geometry," In *proceedings of Seventh International Workshop on Algorithms and Data Structures(WADS2001). Providence, RhodeIsland, USA*, 2001, pp. 165-179.

[12] W. Du and M.J. Atallah. "Privacy-Preserving Cooperative Scientific Computations," In *14th IEEE Computer Security Foundations Workshop, Nova Scotia, Canada*, 2001, pp. 273-282.

[13] W. Du and M.J.Atallah, "Privacy-Preserving Statistical Analysis," In *proceedings of the 17th Annual Computer Security Applications Conference, New Orleans, Louisiana, USA*, 2001, pp. 102-110.

[14] W. Du and M.J. Attalla, "Secure Multiparty Computation Problems and Their Applications: A Review and Open Problems," In *proceedings of new security paradigm workshop, Cloudcroft, New Mexico, USA*,2001, pp. 11-20.

[15] V. Oleshchuk, and V. Zadorozhny, "Secure Multi-Party Computations and Privacy Preservation: Results and Open Problems, 2008. PP.12-18.