Information Security Journal: A Global Perspective
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/uiss20

Secure Multiparty Computation: From Millionaires Problem to Anonymizer

Rashid Sheikh [a] , Durgesh Kumar Mishra [a] & Beerendra Kumar [b]
[a] Acropolis Institute of Technology and Research , Indore, India
[b] Sri Satya Sai Institute of Science and Technology , Sehore, India

Published online: 08 Feb 2011.

Information Security Journal: A Global Perspective, 20:25–33, 2011
Copyright © Taylor & Francis Group, LLC
ISSN: 1939-3555 print / 1939-3547 online
DOI: 10.1080/19393555.2010.544701

Taylor & Francis
Taylor & Francis Group

Secure Multiparty Computation: From Millionaires Problem to Anonymizer

Rashid Sheikh[1], Durgesh Kumar Mishra[1], and Beerendra Kumar[2]
[1]Acropolis Institute of Technology and Research, Indore, India
[2]Sri Satya Sai Institute of Science and Technology, Sehore, India

ABSTRACT In Secure Multiparty Computation (SMC), multiple parties perform joint computation over their private data inputs preserving the privacy of their individual inputs. This type of computation needs to provide correct result while keeping the individual input a secret. In today's scenario of tremendous growth of the Internet and large volumes of online transactions, the concept of data privacy and SMC has become a matter of great concern. People frequently need to perform joint computations for the sake of their mutual benefits, but they are also worried about confidentiality of their private data. This situation arises due to lack of trust among computing parties. For example, two banks may want to find some details of a customer but each of the banks may want to keep their sensitive database a secret or they may not want to disclose the customer's identity. The subject of SMC has evolved from earlier solutions of combinational logic circuits to the recent proposals of anonymity-enabled computation. In this paper, we put together the significant research that has been carried out. We propose new possibilities of problem discovery and its analysis. We put critical issues and challenges and the level of adaptation achieved before the researchers. We also provide some research proposals based on the literature survey.

KEYWORDS secure multiparty computation, privacy, security

INTRODUCTION

The development of the Internet has provided tremendous opportunities for joint computations where each cooperative party wants to evaluate some common function based on its private data input. Such a computation is aimed at correctness of the result and the preservation of privacy of individual data inputs. This computation is called Secure Multiparty Computation (SMC) (Du & Atallah, 2000). This type of computation can occur between two or more untrusted parties. Parties may want to know the result of the evaluation of function of their inputs, but they may not want to disclose their private data to each other. Consider the following real life examples:

Address correspondence to
Rashid Sheikh, Acropolis Institute of
Technology and Research, Magliya
Square, Bypass Road, Indore,
PIN–453771, MP, India.
E-mail: rashidsheikhmrsc@yahoo.com

1. A person wants to know with the help of his DNA pattern about some genetic disease but does not want to disclose his or her DNA pattern or even the result of the query.
2. Mobile phone companies want to know total number of customers in an area but not reveal the number to other companies.
3. All students in a class cooperatively want to know the average of the marks obtained in a particular subject, but no student is willing to disclose individual marks to other students.
4. Four brothers living separately want to compute jointly the total wealth of their family, but no brother is willing to disclose his wealth to the other brother.
5. One bank wants to know the loan details of a customer from another bank, but the requesting bank may not want to disclose the identity of that customer.

Formally, in SMC multiple parties $P_1, P_2, \ldots, P_n$ are involved in computation of some public function of their private inputs $D_1, D_2, \ldots, D_{n,}$ respectively. Each party P_i want to know the common function f $(D_1, D_2, \ldots, D_n)$ without disclosing value of its data D_i to other parties. Many models have been proposed in the literature for the study and analysis of SMC problems. Broadly, two model paradigms are popular:

1. Ideal Model Paradigm of SMC.
2. Real Model Paradigm of SMC.

In the ideal model, there exists some trusted third party (TTP) among participating parties. Parties send their private data inputs to the TTP, which is supposed to perform computation on behalf of these parties. The TTP is assumed to be incorruptible in the sense that it never discloses the private data of one party to another. The TTP, after evaluation of common function, sends the value of the function to all participating parties. The only detail all parties know is the output of computation. Thus the privacy of the input is preserved. In this model, if some party behaves maliciously, the result of the computation may be incorrect because the party may supply invalid input to the TTP but the privacy will be preserved. If the TTP turns corrupt, the privacy may be destroyed. The ideal model of SMC for two parties is shown in Figure 1. The same can be extended for more than two parties. Participating parties provide their private data inputs $x_1, x_2, \ldots, x_n$ to the TTP. The TTP then evaluates some function $f(x_1, x_2, \ldots, x_n)$ and sends back this value to all the participating parties. In a practical scenario, the role of TTP is played by some government or private organization which works as a service provider for this computation. The ideal model of SMC is expensive due to the cost of working of the TTP. One more drawback of this model is that the trustworthiness of the TTP is significant. When TTP turns corrupt, the whole notion of the SMC becomes worthless. However, today this model is frequently used due to easy implementation and use of tools that prevent the TTP from becoming malicious. Traditionally, one TTP is supposed to be sufficient. Recently (Mishra, Koria, Kapoor, & Bahety, 2009) proposed a multiple TTP computation model where a TTP can be selected at run time reducing the probability of data leakage.

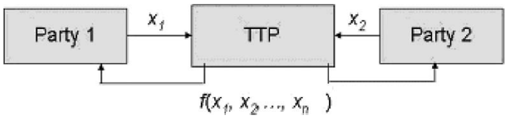

FIGURE 1 Ideal model of SMC.

In the real model there exists no external party which can be trusted for keeping the data a secret. Cooperating parties in this model agree on some protocol which is to be run among themselves for privacy preservation and computation of correct result. This model for two parties is depicted in Figure 2. Parties do not share actual inputs with each other. The values sent by parties are some function of their private data. What exists between parties is a theoretical computation machine. The real model of SMC is said to be secure if an adversary can carry out some attack, which is also possible in the ideal model of SMC. An adversary is a party with malicious intentions. The behavior of the adversary must be properly defined for the universal quantification of the real model of SMC. The adversary's power must be properly assumed. An adversary can be static or adaptive in nature. A static

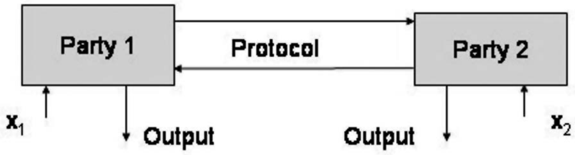

FIGURE 2 Real model of SMC.

adversary is malicious in nature prior to the execution of the protocol. An adaptive adversary turns malicious during the execution of the protocol. A semi-honest adversary follows the protocol but tries to learn something other than the output of the computation. A corrupt or malicious adversary is that which does not follow the steps of the protocol and tries to learn some information other than the result. Different protocols are designed for different nature of the adversary. A protocol designed for semi honest adversary may not work with the corrupt adversary.

Researchers have defined and analyzed many general SMC problems, including privacy-preserving database query, privacy-preserving scientific computation, privacy-preserving intrusion detection, privacy-preserving statistical analysis, privacy-preserving geometric computation, and privacy-preserving data mining. Many real life problems are also studied based on these general problems, for example, privacy-preserving social network analysis, privacy-preserving supply chain management, privacy-preserving auction management, privacy-preserving e-Governance, privacy-preserving private information retrieval, and privacy-preserving signature and face detection. Solutions to the general SMC problems are available (Yao, 1982; Goldreich et al., 1987; Goldreich, 1998). As pointed out by Goldreich (1998), however, special solutions for special SMC problems are more efficient than general solutions.

In the literature, there are three types of solutions available to SMC problems:

1. Randomization methods.
2. Cryptographic techniques.
3. Anonymization methods.

In the randomization method, parties use random numbers for hiding their data and perform computation over hidden data. Protocols are made such that the results of the computation over hidden data are the same as the results of computation over actual data. The best and easily understood example of the randomization SMC method is the secure sum protocol given by Clifton, Kantarcioglu, Vaidya, Lin, and Zhu (2002), which allows multiple parties to compute the sum of their individual data while preserving the privacy of their data. In this protocol, one of the parties is selected as the protocol initiator which selects and adds a random number to its own data. The sum is then transmitted to the next party. The next party simply adds the data to the received sum and then sends this new sum to the next party. This procedure is repeated until the protocol initiator receives the sum of all the data and the random number. Since the random number is known only to the protocol initiator party, it subtracts the random number from the sum and announces the result to all the parties. The architecture of secure sum protocol is depicted in Figure 3. This protocol works well when all parties are honest in the sense that they follow each step agreed in the protocol and never try to know other information except the result of the computation. It is analyzed in secure sum protocol that when two adjacent parties to a middle party turn semi-honest or malicious, they can obtain the middle party's data as well as generate incorrect result. In this protocol, as the number of parties increases the probability of a party getting victimized decreases. Further efforts can be made to decrease the probability of attack over individual data. One way is to break the data into number of segments and then taking the sum of segments in a cumulative manner.

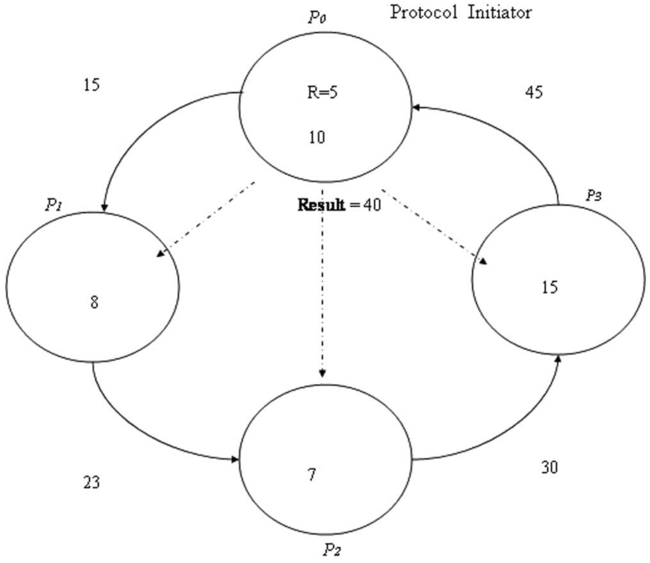

FIGURE 3 Architecture of the secure sum protocol (Clifton et al., 2002).

The cryptographic techniques solutions for SMC use building blocks for secure computation (Oleshchuk & Zadorozhny, 2007). Some of the important building blocks are:

1. Yao's Millionaires Problem.
2. Homomorphic Encryption.
3. Oblivious Transfer.
4. Private Matching.

Yao's Millionaires Problem: The problem allows two parties to compare their private values without disclosing these values to each other (Yao, 1982). Yao provided the solution to this problem using symmetric cryptography. This component for SMC is useful in applications such as online bidding and auctions. Many researchers proposed solutions to this problem (Ioannidis & Grama, 2003; Shindong, Daoshun, Yiqi, & Ping, 2008; Amirbekyan & Estivill-Castro, 2009; Sheikh, Vyas, Kumar, & Mishra, 2009; Cachin, 1999). Cachin used untrusted third party to improve the performance. The same problem can be extended to multiparty case and is useful for the SMC solution. Thus solution to this problem can work as the building block of many SMC problems.

Homomorphic Encryption: Homomorphic encryption is a type of encryption in which some operation is performed on the plaintext and a different operation is performed on the ciphertext. A well-known example of homomorphic encryption is the RSA algorithm where plaintext P is operated as shown in (1) to get the ciphertext C where the pair of the variables (e, n) is encryption key. The plaintext can be reproduced from the ciphertext by performing some different operation as indicated in (2) where the pair of the variables (d, n) is the decryption key.

$$C = P^e \bmod n \tag{1}$$

$$P = C^d \bmod n \tag{2}$$

Many homomorphic systems are proposed in the literature (Benaloh, 1994; Paillier, 1999; Naccache & Stern, 1998). The cryptosystems using homomorphic encryption use modular addition and modular multiplication as the two basic operations. The homomorphic systems have some interesting property. For example, consider an RSA homomorphic system for two plaintexts, x_1 and x_2. Their encryption will result in two ciphertexts as: E(x_1) = x_1^{e} mod n and E(x_2) = x_2^{e} mod n . The product of these two ciphertexts is E(x_1). E(x_2) = $(x_{1.}\ x_2)^{\mathrm{e}}$ mod n which gives the property as E(x_1). E(x_2) = E($x_{1.}\ x_2$). In other words, the product of the ciphertexts is same as the ciphertext of the product of the plaintexts.

Oblivious Transfer: Oblivious transfer is a type of information transfer between a sender and receiver in which the sender is unaware of the value received by the receiver and the receiver is unaware of the position of the value at the sender side. The oblivious transfer is a main component of the SMC. With the help of implementation of the oblivious transfer, we can obtain any secure computation (Kilian, 1988). Kilian showed that oblivious transfer is sufficient to perform any secure computation. The first oblivious transfer was suggested by Rabin (1981). Later, Even, Goldreich, and Lempel (1985) proposed the same protocol as one-out-of-two oblivious transfer (1-2-OT). This is depicted in the Figure 4. The sender sends {m_0, m_1} ∈ {0, 1}. The receiver sends i ∈ {0, 1} and the oblivious transfer sends m_i to the receiver. The sender is unaware about i and the receiver is unaware about which of the inputs it received. The natural generalization of 1-2-OT is the 1-n-OT where the sender sends n messages and the receiver receives any one of these without knowing the senders sequence of the messages. The sender is unaware about a particular message received by the receiver.

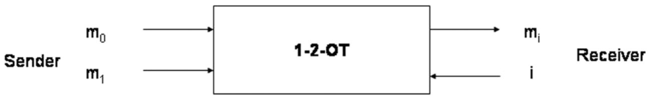

FIGURE 4 One-of-two-oblivious transfer.

Private Matching: Private matching allows two parties to find the intersection of their data sets without disclosing their private data sets to each other. This building block is useful when performing secure computations over databases. The protocols used for private matching use the properties of homomorphic encryption. Different protocols are available for semi honest parties and that for malicious parties (Freedman, Nissim, & Pinkas, 2004; Agrawal, Evfimievski, & Srikant, 2003).

In anonymization method, the identities of the parties are hidden by different techniques. This method has been proposed with the ideal model of SMC where a TTP accepts inputs from parties, evaluates the function of these inputs and sends result of computation to each of the parties. The identity of the party can be made ambiguous as proposed by Mishra and Chandwani (2007). They proposed two protocols for hiding the identity of cooperating parties. The architecture of the *anonypro* protocol is shown in Figure 5 where anonymizers A_1, A_2, . . ., A_n are used between parties and the TTP. As shown in the Figure 5, the architecture has three layers: the lowest layer is the input layer where parties send inputs for computation,

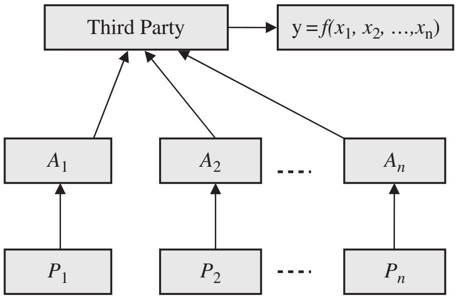

FIGURE 5 Architecture of *anonypro* protocol (Mishra & Chandwani, 2007).

the middle layer is the security layer where anonymizers hide the identity of parties, and the topmost layer is the computation layer where TTP computes the desired function and sends the result to the parties. Since the TTP is oblivious about the party, no threat exists in case TTP turns malicious. In this protocol, the anonymizers needs to be honest. In another protocol, *extended anonypro* four layers are used as shown in Figure 6. The new layer is used to break the data of the individual party into number of segments. Each party selects one of multiple anonymizers. Since an anonymizer can learn only piece of data, the privacy of the data is maintained in case an anonymizer turns malicious.

The motive of this paper is to put significant research work done by the researchers since the beginning of the subject of the SMC before future researchers and then to propose some modification to selected problems. These modifications are developed based on our exhaustive survey of the literature of the SMC and some observation of the architecture and the results of some methods. These research proposals can lead future researchers to develop efficient and architectures with more secure protocols.

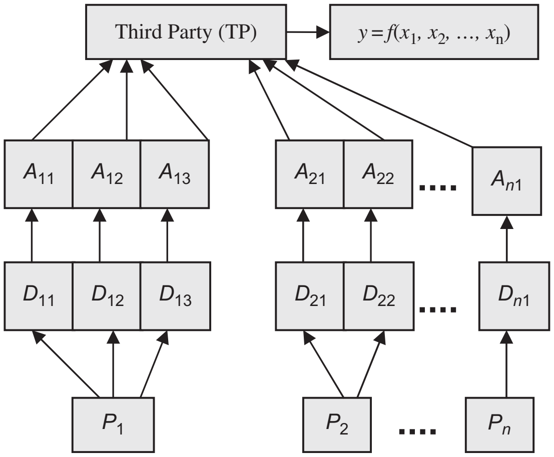

FIGURE 6 Architecture of *extended anonypro* protocol (Mishra & Chandwani, 2007).

RELATED WORK

The history of SMC concept started with Yao's Millionaires problem (Yao, 1982) when he proposed the problem as: How can two millionaires know who is richer without disclosing their individual wealth to each other? He provided a solution to this problem using cryptographic techniques. Later, many researchers worked out on the same concept and extended the idea from two party computations to multiparty computation (Goldreich, Micali, & Wigderson, 1987). Before Yao, Diffie and Hellman (1976) provided background of computational cryptography and helped much of the work in SMC. They proposed a secure protocol for key exchange based on the computational difficulty in the discrete logarithm function. Shamir (1979) introduced the problem of secret sharing which became an important application domain for SMC.

The early solutions to SMC problems used combinational logic circuit where each party cooperatively runs some protocol to provide input to each logic gate in the combinational logic circuit. For example, a simple hardware solution to the Yao's millionaires problem can be a digital comparator which accepts two binary numbers as inputs and tells in the output which one is greater (Sheikh, 2009). The block diagram of the digital comparator is shown in Figure 7. The digital comparator accepts two values as binary inputs, compares these two values, and gives result as to which is greater or if both are equal. If in the output we get $E = 1$, then it indicates that both the values are equal. If $G = 1$ then it shows that binary number 1 is greater than binary number 2. If $L = 1$ then it points out that binary number 1 is less than binary number 2. If input lines are made secure and only the values of the output lines are shown to the parties, than the result of

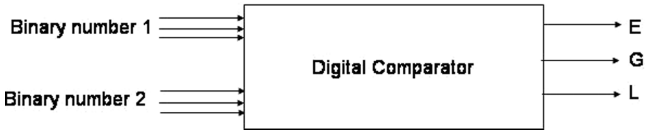

FIGURE 7 Block diagram of the digital comparator.

the comparison is known to preserve the privacy of the individual inputs.

The circuit evaluation protocols are general and simple, but their performance depends on the size of the input. Many specific solutions are proposed for specific SMC problems.

SPECIFIC SMC PROBLEMS

Many specific SMC problems and their solutions are devised by the researchers. Here we present most significant problems and the work done for their solutions.

Private Information Retrieval

This problem consists of a client and a server where the client requests the server to supply the i^{th} bit of the binary sequence of the server without the server knowing anything about i. This problem was introduced by Chor, Kushilevitz, Goldreich, and Sudan (1995).The problem was extended by Gertner, Ishai, Kushilevitz, and Malkin (1998) as the Symmetrically Private Information Retrieval (SPIR), which ensures privacy of the database as well as the privacy of users. The client is also not aware of the bit sequence of the server. Many solutions are proposed for the PIR SMC problem focusing on reducing the communication cost (Chor & Gilbao, 1997; Ishai & Kushilevitz, 1999; Di-Crescenzo, Ishai, & Ostrovsky, 1998; Kushilevitz & Ostrovsky, 1997; Cachin, Micali, & Stadler, 1999; Gertner, Goldwasser, & Malkin, 1998).

Selective Private Function Evaluation

This problem was proposed by Canetti, Ishai, Kumar, Reiter, Rubinfeld, and Wright (2001) as: how a client can evaluate a function $f(x_{i1}, x_{i2}, \ldots, x_{ik})$ with a database $x_1, x_2, \ldots, x_n$ held by many servers by selecting one or more servers and sending the indices $i_1, i_2, \ldots, i_k$ such that the servers know nothing about the indices. This problem is useful when the database to be computed contain some information that must not be revealed to others while doing certain computation over the data sets.

Privacy-Preserving Data Mining

Recently, researchers proposed two different types of the privacy-preserving data mining problems. Lindell and Pinkas (2000) defined the privacy-preserving data mining problem as how two parties can perform a data mining operation on the union of their private databases without disclosing their individual databases to each other or to any third party. They used Iterative Dichotomiser 3 (ID3) algorithm, which allows for building the decision tree without knowing the exact contents of the data records. Agrawal and Srikant (2000) defined the problem as how one party can be allowed to perform data mining operation on the private database of another party without the first party knowing any details of the database of the second party. They used data perturbation method to solve the problem.

Privacy-Preserving Cooperative Scientific Computation

Many organizations may jointly work on a project for their mutual benefit. Each of the organizations may have its own requirements represented as a set of linear system of equations; linear least squares problems or linear programming problems. These are the modern modeling techniques by which different strategies of the planning, routing, scheduling, assignment, and designs are made. Diverse kinds of industries use these models. When a joint venture of many organizations is working on a single project, they need to solve these equations cooperatively. If there is a lack of trust between the parties, however, how could they do so? This problem is evolved as the privacy-preserving scientific computation and can be formulated as: Party 1 has a set of linear equations $M_1 x = a_1$ and party 2 has a set of linear equations $M_2 x = a_2$ where x is n dimensional vector. How these two parties can solve these equations without disclosing their private equations to one another. Solutions to such problems are provided by (Du & Atallah, 2001).

Privacy-Preserving Database Query

The privacy-preserving database query problem can be defined as: one party has a string s and another party has database of the strings $D = \{s_{1,} s_2, \ldots, s_n \}$. How the first party can know whether the database D contains some string s_i that match with s such that the second party does not know the exact string s and the first party does not know the exact database. The match of the strings could be an exact match or an approximate match. The same problem can be used with image template matching (Gonzalezi & Woods, 1992; Jain, 1989). Some research results for remote database access with approximate matching are provided by Du and Atallah (2000).

Privacy-Preserving Geometric Computation

Different types of geometric computation problems are useful in different situations. The *intersection* problem can be formulated as how two parties having their private shapes can know whether these shapes overlap with each other without the parties disclosing their private shapes to one another or even to a third party. If these shapes overlap, nobody should learn where this intersection occurred, for example, when two companies want to set up new business in certain regions but do not want to open business in the same regions. Intersection problem will let them know whether their areas overlap. Another important geometric computation problem is the *point-inclusion,* which can be formulated as one party has a point and another party has a private polygon. How can these parties know whether the point is inside or the outside of the polygon without providing their private information to one another? Nobody is allowed to know the relative positions of the point and the polygon. Another important geometric computation problem is the *range searching* in which one party has a private range and another party has N points. How can these parties know how many points are there in the range without revealing their private information to one another? The c*losest pair* problem allows two parties having their private points to know about the two points which have minimum distance. Solutions for point inclusion problem and intersection problem are provided by Atallah and Du (2001).

Privacy-Preserving Intrusion Detection

Privacy-preserving profile matching problem for intrusion detection can be formulated as one party holds a private database containing a known hacker's behaviors and another party has collected a recent hacker's behavior. How can the second party match the collected profile with the database of the first party without disclosing the actual behavior to the first party? The first party also does not want to disclose its database. Today these techniques are used by many banks for cooperative intrusion detection. (Biskup & Flegel, 2000, 2000a) proposed pseudonym techniques for intrusion detection.

Privacy-Preserving Statistical Analysis

Privacy-preserving statistical analysis over two private data sets of two parties can be defined as how these two parties can find out correlation coefficient and regression line without revealing their private data sets to one another. The correlation coefficient between the two data sets tells about the degree to which the larger values of the variables of one data set go with the larger values of the variables of other data set, and the smaller values of the variables of one data set go with smaller values of other variables of the other data set. The regression line is an equation which provides values of the variables of one data set for given values of the variables of other data set. This analysis is used in predictions. Du et al. proposed solutions for many privacy-preserving statistical problems (Du & Atallah, 2001a). Many other privacy-preserving statistical analysis problems and solutions are available in the literature (Warner, 1965, 1971; Goodstadt & Gruson, 1975; Pollock & Bek, 1976).

OUR RESEARCH CONTRIBUTION FOR SMC

During our literature survey, we were motivated by the work of Clifton et al. (2002), which provided a detailed discussion of the components for SMC toolkit. Secure sum computation was proposed as the important component for solution to SMC problem. In their paper they proposed a secure sum protocol using random numbers. The protocol works well when all parties are honest or semi-honest and no party colludes with another party to know the secret data of some other party. The vulnerability of this protocol lies in the fact that any two colluding neighbors can easily know the secret data of the middle party by simple comparisons of what they send or receive. In order to improve the security of the secure sum computation, we used the segmentation approach and provided a novel protocol *k-secure sum* protocol (Sheikh, Kumar, & Mishra, 2009a). In this protocol, the cooperating parties are arranged in a unidirectional ring, and each party breaks its data block into number of segments. The protocol performs summation for each of the segments separately. We showed that the probability of data leakage by two colluding parties is significantly reduced. The probability is a function of the number of the segments of a data block and is further reduced when the number of the segments is increased. In the same paper (Sheikh et al., 2009a), we proposed an *extended k-secure sum* protocol which used segmentation with randomization to make the protocol more secure against colluding neighbors. In this protocol, each segment summation round uses a random number. In another novel protocol called *ck-secure sum* protocol (Sheikh, Kumar, & Mishra, 2010), we allow parties to change their position in each round of the segment computation so that a particular party does not have same neighbors for all rounds of the segment computation. This protocol guarantees that the colluding neighbors cannot learn all the segments of the data block of a middle party. Based on our segmentation approach we proposed a protocol *k-secure product* for computing the secure product of data of cooperating parties (Sheikh, Joshi, & Mishra, 2009).

DISCUSSION

Based on our literature survey and research contribution in SMC, we provide some guidelines for future researchers. The analysis of the *ck-secure sum* protocol shows that when more than two parties collude with each other there exists a nonzero probability of the data leakage. The protocol could be modified to provide a zero probability when more than two parties collude. The *anonypro* and *extended anonypro* protocols (Mishra & Chandwani, 2007) use a third party for the computation of the common function. The architecture of these protocols can be modified to minimize the cost of the computation. A simple hardware implementation of the Yao's millionaire problem (Sheikh et al., 2009) is suitable for integer values only. The same implementation can be performed for the real values. The original solution of the Yao's millionaires' problem (Yao, 1982) was for the known range of values, which also can be designed for unknown ranges. The secure sum protocols proposed in 2009 and 2010 for single data block values (Sheikh et al., 2009, 2010) can be extended for the vector data. These protocols are suitable for semi-honest parties. The same can be designed for the corrupt parties.

CONCLUSION

The subject of the SMC is so important and relevant that it will soon become an integral part of our computing environment. The subject has evolved from the early millionaire's problem to the recent proposals of the anonymity enabled protocols. Many specific SMC problems are studied and efforts are continuously going on to find new areas of the SMC. Based on the specific problems, many real life problems are also studied. The goal behind all these studies is to achieve privacy of the input and get the correct result with minimum communication and the computation cost. In this paper we presented the important research work done in SMC and its components. We also presented our research contribution for developing secure sum computation protocols, which is an important component of the SMC solution. Finally, we proposed new research possibilities based on our literature survey and our research contribution. These proposals will work as guidelines for future researchers.

REFERENCES

Agrawal, R., Evfimievski, A., and Srikant, R. (2003). Information sharing across private databases. ACM SIGMOD, *International Conference on Management of Data*, San Diego, CA. June 10–12.

Agrawal, R. and Srikant, R. (2000). Privacy-preserving data mining. *Proceedings of the 2000 ACM SIGMOD on Management of Data*, pp. 439–450.

Amirbekyan, A. and Estivill-Castro, V. (2009). Practical protocol for Yao's millionaires problem enables secure multi-party computation of metrics and efficient privacy-preserving k-NN for large data sets. *Knowledge and Information Systems*. Available from: http://dx.doi.org/10.1007/s10115-009-0233-z

Atallah, M.J. and Du, W. (2001). Secure multi-party computational geometry. *WADS2001: Seventh International Workshop on Algorithms and Data Structures*, pp. 165–179.

Benaloh, J. (1994). Dense probabilistic encryption. *Proceedings of the Workshop on Selected Areas of Cryptography*, pp. 120–128.

Biskup, J. and Flegel, U. (2000). On pseudonymization of audit data for intrusion detection. *Workshop on Design Issues in Anonymity and Unobservability*, pp. 161–180.

Biskup, J. and Flegel, U. (2000a). Transaction-based pseudonyms in audit data for privacy respecting intrusion detection. In Debar, H., Me, L. and Wu, S. F. (Eds) Recent Advances in Intrusion Detection, pp. 28–48.

Cachin, C. (1999). Efficient private bidding and auctions with an oblivious third party. *Proceedings of the 6th ACM Conference on Computer and Communications Security*, pp. 120–127.

Cachin, C., Micali, S., and Stadler, M. (1999). Computationally private information retrieval with polylogarithmic communication. Advances in Cryptology: EUROCRYPT '99, Lecture Notes in Computer Science, 1592, 402–414.

Canetti, R., Ishai, Y., Kumar, R., Reiter, M.K., Rubinfeld, R., and Wright, R.N. (2001). Selective private function evaluation with applications to private statistics. *Proceedings of the Twentieth Annual ACM Symposium on Principles of Distributed Computing, PODC '01*, pp. 293–304.

Chor, B., Kushilevitz, E., Goldreich, O., and Sudan, M. (1995). Private information retrieval. *Proceedings of the 36th Annual IEEE Symposium on Foundations of Computer Science*, pp. 41–50.

Chor, B. and Gilbao, N. (1997). Computationally private information retrieval (extended abstract). *Proceedings of 29th Annual ACM Symposium on Theory of Computing*, pp. 304–313.

Clifton, C., Kantarcioglu, M., Vaidya, Lin X., and Zhu, M.Y. (2002, December). Tools for privacy-preserving distributed data mining. J. SIGKDD Explorations, Newsletter 4(2), 28–34.

Diffie, W. and Hellman, M. (1976). New directions in cryptography. Information Theory, IEEE Transactions, 22(6), 644–654.

Di-Crescenzo, G., Ishai, Y., and Ostrovsky, R. (1998). Universal service-providers for database private information retrieval. *Proceedings of the 17th Annual ACM Symposium on Principles of Distributed Computing*, pp. 91–100.

Du, W. and Atallah, M.J. (2001a). Privacy-preserving statistical analysis. *Proceedings of the 17th Annual Computer Security Applications Conference*, pp. 102–110.

Du, W. and Atallah, M.J. (2001). Privacy-preserving cooperative scientific computations. *14th IEEE Computer Security Foundations Workshop*, pp. 273–282.

Du, W. and Atallah, M.J. (2000). Protocols for secure remote database access with approximate matching. *7th ACM Conference on Computer and Communications Security (ACMCCS 2000)*, the First Workshop on Security and Privacy in E-commerce, Athens, Greece.

Du, W. and Atallah, M.J. (2000). Secure multiparty computation problems and their applications: A review and open problems. *Proceedings of New Security Paradigms Workshop*, pp. 11–20.

Even, S., Goldreich, O., and Lempel, A. (1985). A randomized protocol for signing contracts. Commutations of ACM, 28(6), 637–647.

Freedman, M., Nissim, K., and Pinkas, B. (2004). Efficient private matching and set intersection. *Advances in Cryptology Eurocrypt '2004 Proceedings*, pp. 1–19.

Gertner, Y., Ishai, Y., Kushilevitz, E., and Malkin, T. (1998). Protecting data privacy in information retrieval schemes. *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, STOC '98*, pp. 151–160.

Gertner, Y., Goldwasser, S., and Malkin, T. (1998). A random server model for private information retrieval. *2nd International Workshop on Randomization and Approximation Techniques in Computer Science (RANDOM '98)*. Barcelona, Spain, October. Springer.

Goldreich, O., Micali, S., and Wigderson, A. (1987). How to play any mental game. *STOC '87: Proceedings of the Nineteenth Annual ACM Conference on Theory of Computing*, pp. 218–229.

Goldreich, O. (1998). Secure multi-party computation (working draft). Available from: http:/www.wisdom.weizmann.ac.il/home/oded/public html/foc.html

Gonzalezi, R. and Woods, R. (1992). Digital image processing. Reading, MA: Addison-Wesley.

Goodstadt, M. S. and Gruson, V. (1975, December). The randomized response technique: A test on drug use. Journal of the American Statistical Association, 70(352), 814–818.

Ioannidis, I. and Grama, A. (2003). An efficient protocol for Yao's millionaires problem. *Proceedings of the 36th Hawaii International Conference on System Sciences*, pp. 6–9.

Ishai, Y. and Kushilevitz, E. (1999). Improved upper bounds on information-theoretic private information retrieval (extended abstract). *Proceedings of the Thirty-first Annual ACM Symposium on Theory of Computing*, pp. 79–88.

Jain, A. (1989). Fundamentals of digital image processing. Englewood Cliffs, NJ: Prentice Hall.

Kilian, J. (1988). Founding cryptography on oblivious transfer. *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing, STOC '88*, pp. 20–31.

Kushilevitz, E. and Ostrovsky, R. (1997). Replication is not needed: Single database, computationally private information retrieval. *Proceedings of the 38th Annual IEEE Computer Society Conference on Foundation of Computer Science*, pp. 20–22.

Lindell, Y. and Pinkas, B. (2000). Privacy preserving data mining. In Advances in Cryptology – Crypto2000, Lecture Notes in Computer Science, 1880.

Mishra, D.K., Koria, N., Kapoor, N., and Bahety, R. (2009). A secure multi-party computation protocol for malicious computation prevention for preserving privacy during data mining. International Journal of Computer Science and Information Security, 3(1), 1–6.

Mishra, D.K. and Chandwani, M. (2007, February). Extended protocol for secure multiparty computation using ambiguous identity. WSEAS Transaction on Computer Research, 2(2), 227–233.

Naccache, D. and Stern, J. (1998). A new public key cryptosystem based on higher residues. *Proceedings of the 5th ACM Conference on Computer and Communications Security, CCS '98*, pp. 59–66.

Oleshchuk, V. and Zadorozhny, V. (2007). Secure multi-party computations and privacy preservation: Results and open problems. Telektronikk: Telenor's Journal of Technology, 103(2), 20–26.

Paillier, P. (1999). Public-key cryptosystems based on composite degree residuosity classes. EUROCRYPT'99, Lecture Notes in Computer Science, 223–238.

Pollock, K.H. and Bek, Y. (1976, December). A comparison of three randomized response models for quantitative data. Journal of the American Statistical Association, 71(356), 994–886.

Rabin, M.O. (1981). How to exchange secrets by oblivious transfer. Technical Report TR-81, Aiken Computation Laboratory, Harvard University.

Shamir, A. (1979). How to share a secret. Commun. ACM, 22(11), 612–613.

Sheikh, R., Vyas, M., Kumar, B., and Mishra, D.K. (2009). A simple hardware implementation of Yao's millionaires problem. *Communicated to a National Conference*. In the proceedings of Third CSI National Conference on Education and Research (ConFER2010), Guna, India, March, 303–308.

Sheikh, R., Kumar, B., and Mishra, D.K. (2009a, November). Privacy-preserving k-secure sum protocol. International Journal of Computer Science and Information Security, 6(2), 184–188.

Sheikh, R. (2009). Digital computers: Electronics, organization and fundamentals, 9th edition. India: *Nakoda Publishers and Printers*, pp. 281–283.

Sheikh, R., Joshi, M., and Mishra, D.K. (2009). A protocol for computing product while preserving privacy of inputs. *Second Bhartiya Vigyan Sammelan Conference*, Indore, India.

Sheikh, R., Kumar, B., and Mishra, D.K. (2010, January). Changing neighbors k-secure sum protocol for secure multi-party computation. International Journal of Computer Science and Information Security, 7(1), 239–243.

Shindong, L., Daoshun, W., Yiqi, D., and Ping, L. (2008, January). Symmetric cryptography solution to Yao's millionaire's problem and an evaluation of secure multiparty computations. International Journal of Information Sciences, 178(1), 244–255.

Warner, S.L. (1971, December). Randomized response: A survey technique for eliminating evasive answer bias. Journal of the American Statistical Association, 66(336), 884–888.

Warner, S.L. (1965, March). Randomized response: A survey technique for eliminating evasive answer bias. Journal of the American Statistical Association, 60(309), 63–69.

Yao, A.C. (1982). Protocol for secure computations. *Proceedings of the 23rd Annual IEEE Symposium on Foundation of Computer Science*, Indore, India, pp. 160–164.