Cryptography for Privacy-Preserving Data Mining

Fuad Hamidli

Middle East Technical University, Department of Cryptography, Ankara, Turkey
fuad_hamidli@yahoo.com

***Abstract*— With the increasing amount of data and using huge databases in information systems it becomes very important to develop methods between two or more distributed systems in order to conduct joint work (research). Data mining is such a method which extracts the useful information from the large databases. Privacy problems occur day by day, as the result of increasing technologies using internet (cloud), or joint work between two or more companies using distributed data-mining algorithms which have huge databases. When data mining techniques are used in a malicious way, classical methods are not sufficient to provide adequate privacy. In this paper, we review security mechanisms and some cryptographic techniques in order to solve privacy problems in structures using data-mining algorithms.**

Keywords*— *privacy; data mining; secure two-party computation; cryptography

I. INTRODUCTION

Privacy is one of the most important issue in information systems . Especially, when information systems need to share information among different untrusted entities the protection of information becomes essential. The more huge databases increase in today's world, the more security and privacy gaps occur in the system when there is a sharing of information between the parties. Privacy should be thought carefully when two or more parties having confidential databases wish to run a data mining algorithm on the union of their databases without revealing any unnecessary information. For example, consider medical institutions (two or more separate) that want to conduct joint survey without revealing their patients privacy. In this case it is essential to protect the sensitive information but it is also required to enable its use for the future researches. In particular, although the parties know that combining their data has mutual benefits for them, none of them is willing to capture its database to any other party. There are also many examples that privacy of companies (citizens) faces with problems as the usage of internet like structures are increasing day by day. These problems become prevalent when using cloud-like structures. Service providers (including the most famous ones in social media) have made a business out of exploiting this data for advertising purposes. Large data is collected about habits, preferences of the users and is used to send appropriate advertisements to those users or to sell these information to other companies. All these problems are not due to a lack of available security mechanisms. Cryptography provides available protocols, encryption algorithms in order to solve security problems.

II. PRELIMINARIES, DEFINITIONS

A. *Semi-honest model*

In cryptography, there are different adversarial behaviors. Typically we consider two of them: semi-honest adversary and malicious adversary. In semi-honest model, (also known as passive) one of the (or both of them) party correctly follows the protocol but attempts to learn additional information by analyzing the transcript of messages during the protocol execution. Security in the presence of semi-honest adversaries provides only a weak security guarantee, and is not adequate in many settings.

B. *Malicious model*

In malicious model, on the other hand, one of the parties may arbitrarily deviate from the protocol. Furthermore, he may lie about his inputs, may quit at any points. For example, assume that there is a protocol between two parties and consider a step that wants to choose random number from both parties. If the party is semi honest, we can believe that the chosen number is indeed random. If the party is malicious, then he might choose number not randomly in order to gain extra knowledge about other party [10].

Therefore, in cryptography common approach is to first design protocol in semi-honest model and then to convert it to a malicious model. The protocols that are secure in the malicious model provide a very strong security guarantee as honest parties are protected irrespective of the adversarial behavior of the corrupted parties.

C. *Ideal (real) model*

In ideal model each party sends their inputs to the trusted third party who computes the function for them. In real model, parties run a protocol without trusted help. In fact, a protocol is secure if any attack on a real protocol can be carried out in the ideal model.

D. *Security*

A protocol P securely computes a function f if:

- For every real-model adversary A there exists an ideal model adversary S such that the result of a real execution of P with A is indistinguishable from the result of an ideal execution with S. (where the trusted party computes f)

III. SECURITY MECHANISMS AND CRYPTOGRAPHIC TECHNIQUES

In data mining structures, we have a problem of function evaluation between separate parties with their own inputs without revealing any other information. Here, we describe some techniques and protocols in cryptography that help to solve secure function evaluation problem.

A. Oblivious Transfer (OT)

Oblivious transfer is one of the main building blocks of secure computation. A simple cryptographic primitive first was introduced by Rabin in 1981[1]. It was shown by Kilian [2] that oblivious transfer is sufficient for the secure computation in the sense that given an implementation of oblivious transfer , and no other cryptographic primitive, one could construct secure function evaluation protocol. In fact, one can essentially base any cryptographic protocol on this primitive. In oblivious transfer, we have two main aspects, assuming that Alice sends a message to Bob:

- We want that Bob receives the message with probability ½
- We also want that Alice does not know whether Bob receives it or not.

As an alternative to the Rabin OT, Even, Goldreich and Lempel suggested 1-out-2 oblivious transfer notion [3]. In this protocol, there are two parties; sender and receiver. The sender's input is (X_0, X_1) and the receiver's input is a bit $\sigma \in \{0,1\}$. At the end of the protocol the receiver learns X_σ (and nothing else) and the sender learns nothing. To make clear, if I stands for inputs and O stands for output, the oblivious transfer is the function from (I_A, I_B) to (O_A, O_B) that takes $((X_0, X_1),\sigma)$ to (λ, X_σ) where λ is the empty output.

By using public key cryptography it is known how to design protocol based on oblivious transfer. In [3] and [4] one can find efficient and simple OT protocols in the case semi-honest model. One example is that, receiver generates two random public keys, P_σ and $P_{1-\sigma}$ and he knows the decryption key of P_σ but does not know the decryption key of $P_{1-\sigma}$. Receiver sends these keys to the sender. Sender encrypts X_0 with the key P_0 and encrypts X_1 with the key P_1, and sends these encryptions to the receiver. The receiver can decrypt X_σ but not $X_{1-\sigma}$. It is obvious that the sender does not learn anything about σ , because the only message she receives are two random public keys and he cannot find which one of them has private key that is chosen by receiver. For the sender's privacy, since receiver knows only one private key he can decrypt only one of the inputs. In the malicious case, oblivious transfer should ensure that receiver chooses public keys suitably. This can be done by zero knowledge proofs which will be used by receiver to prove that he chooses right keys. For more efficient proofs one can read [5].

1-out of-2 OT can be extended to the 1-out of –n OT. In fact, we can assume that $n=2^m$. Protocol can be designed as:

- Alice chooses 2m random numbers k_1, k'_1, k_2, $k'_2,\ldots,k'_m$
- For each message m_i, for each j: if the j th bit of i is 0, then the message is XOR'ed by k_i, otherwise the message is XOR'ed by k'_I
- For example, i =1011 then m_i is XOR'ed by k_1 XOR'ed by k'_2 XOR'ed by k_3 XOR'ed by k_4
- For each j, Alice and Bob run a 1-out of-2 protocol such that Bob learns one of the two random numbers k_j and k'_j
- The random numbers that Bob learns can only help him to learn one message
- Alice clearly cannot learn Bob's choice.

B. Oblivious polynomial evaluation

There is another protocol that solves oblivious polynomial evaluation, which has sender and receiver. The sender's input is a polynomial P of degree k over some finite field F and receiver's input is an element z in F. (the degree k is publicly known). The protocol is that the receiver learns P(z) and nothing else and sender learns nothing. This problem was introduced in [6] and was solved in an efficient way

C. Secure two-party computation

Let Alice (A) and Bob (B) be two parties with inputs x and y respectively. Suppose that they wish to compute a function f(x,y) without revealing their inputs each. Yao presented a solution to this problem by designing protocol based on combinatorial garbled circuits with gates over some fixed basis B. [7] This garbled circuit is an encrypted form of the function $f(x,y)=(f_1(x,y), f_2(x,y))$ where $f_1(x,y)$ and $f_2(x,y)$ denotes respective outputs of the two parties and jointly computed by the two parties.

Yao's garbled circuit is shortly as follows:

The idea is that Bob generates the garbled circuits and Alice evaluates the garbled circuits. Assume that the function f(x,y) is represented as Boolean circuit. For each wire Bob in the Boolean circuit Bob uses two random bit strings that are assigned to 0 and 1 respectively. Bob sends only the garbled strings corresponding to his input values to Alice. Here Bob uses 1-out of -2 oblivious transfers without learning which strings Alice gets. At the end of the evaluation, Alice needs to tell Bob the garbled strings she found for his output wires and Alice converts to garbled circuit to bits. We refer [7] for more detailed description of protocol.

This protocol works for any probabilistic polynomial-time function. There are some recent protocols (and the name of the journals, conferences) based on Yao's garbled circuit:

- [EUROCRYPT07] B.Pinkas and Y.Lindell
- [Journal of Cryptology] Y.Lindell and B.Pinkas. A proof of security of Yao's protocol for two party computation.2009
- [CANS09] V.Kolesnikov,A.-R. Sadeghi, T. Schneider. Improved garbled circuit building blocks and applications to auctions and computing minima.

- [EUROCRYPT11] A.Shelat and C.-H.Shen. Two output secure computation with malicious adversaries.
- [EUROCRYPT11] Bendlin.,R.,Damgard.,I.,Orlandi, C.,Zakarias, S. Semi-homomorphic encryption and multiparty computation.

D. The multi-party case

In the multi-party case, there are protocols that enable parties to compute any joint function of their inputs without revealing any other information about the inputs. In fact, this is to compute the function with the same privacy as in the ideal model. There are several constructions for this case (for more information [8]) which computes f as a circuit and evaluates.

CONCLUSION

In conclusion, in data mining systems the usage of cryptographic techniques as above increases the security. For example, there is a database algorithm ID3 which is used to construct decision trees. In [9] there is an efficient privacy preserving distributed computation protocol for ID3 based on Yao's garbled circuit. There are many applicable areas of privacy computation protocols as data mining. Some countries using smart-meter for electricity (such as USA, UK etc.) also use these cryptographic techniques to keep privacy of citizens safe. Thus, to guarantee privacy in data-mining systems, the use of cryptography properly is inevitable.

REFERENCES

[1] M.O.Rabin ,"How to exchange secrets by oblivious transfer", Technical Memo TR-81,Aiken Computation Labaratory, 1981.

[2] J. Kilian, "Founding cryptography on oblivious transfer," ACM STOC'88, pp 20-31, 1988.

[3] S.Even, O. Goldreich, A.Lempel, "A randomized protocol for signing contracts," Communications of the ACM, vol. 28, pp. 637-647,1985.

[4] O.Goldreich, "Secure multi-party computation," Manuscript, 2002. Available at http://www.wisdom.weizmann.ac.il/oded/pp.html

[5] M.Naor and B.Pinkas, "Efficient oblivious transfer protocols," Proceedings of 12th SIAM Symposium on Discrete Algorithms(SODA), January 7-9 2001,Washington DC, pp-448-457.

[6] M.Naor and B.Pinkas, "Oblivious transfer and polynomial evaluation," Proceedings of the 31th Annual Symposium on the Theory of Computing (STOC), ACM, 1999, pp 245-254.

[7] A.C.Yao, "How to generate and exchange secrets," Proceedings 27th Symposium on Foundations of Computer Science (FOCS), IEEE,1986 pp 162-167.

[8] O.Goldreich ,S.Micali and A.Wigderson, "How to play any mental game - A completeness theorem for protocols with honest majority," Proceedings of the 19th Annual Symposium on the Theory of Computing (STOC) , ACM,1987, pp 218-229.

[9] Y.Lindell and B.Pinkas, "Privacy preserving data mining," Journal of Cryptology, Vol.15, No. 3, pp. 177-206, 2002.

[10] B.Pinkas, "Cryptographic techniques for prviacy–preserving data mining," ACM SIGKDD Explorations Newsletter,vol. 4, no. 2, pp. 12-19, 2002.