Information Theoretical Analysis of Two-Party Secret Computation

Da-Wei Wang*, Churn-Jung Liau**, Yi-Ting Chiang, and Tsan-sheng Hsu***

Institute of Information Science Academia Sinica, Taipei, 115, Taiwan
{wdw,liaucj,ytc,tshsu}@iis.sinica.edu.tw

Abstract. Privacy protection has become one of the most important issues in the information era. Consequently, many protocols have been developed to achieve the goal of accomplishing a computational task cooperatively without revealing the participants' private data. Practical protocols, however, do not guarantee perfect privacy protection, as some degree of privacy leakage is allowed so that resources can be used efficiently, e.g., the number of random bits required and the computation time. A metric for measuring the degree of information leakage based on an information theoretical framework was proposed in [2]. Based on that formal framework, we present a lower bound of the scalar product problem in this paper, and show that to solve the problem without the help of a third party, approximately half the private information must be revealed. To better capture our intuition about the secrecy of various protocols, we propose two more measurements: *evenness* and *spread*. The first measures how evenly the information leakage is distributed among the participants' private inputs. The second measures the size of the smallest set an adversary could use to obtain the same ratio of leaked information that could be derived in the worst case scenario.

Keywords: Privacy Analysis, Private Computation, Scalar Product

* Corresponding Author. Joint appointment faculty member of National Yang Ming University Taiwan. Supported in part by NSC (Taiwan) grant 94-2213-E-001-030.
** Supported in part by NSC(Taiwan) grant 94-2213-E-001 -007
*** Supported in part by NSC (Taiwan) grant 94-2213-E-001-014.

1 Introduction

Privacy protection is one of the most pressing issues in the information era. The massive databases spread over the Internet are gold mines for some and, at the same time, one of the greatest threats to privacy for others. How to accomplish a computational task cooperatively without revealing the participants' private inputs has therefore gained a great deal of attention in recent years and the development of efficient solutions is now an active research area. In theory [11, 7], it is possible to securely compute almost any function without revealing anything, except the output. Unfortunately, the theoretical results are not readily applicable to real applications due to their high computational complexity.

Most theoretical approaches adopt a computationally indistinguishable view of secrecy and try to find provable secure solutions, but such a definition leaves little room to quantify secrecy. Meanwhile, in application-oriented studies, researchers usually take an intuitive approach to the definition of secrecy and try to argue for the secrecy of protocols by refuting possible attacks. There is a gap between these two approaches in terms of provable secrecy. Although, privacy is a basic human right, it is not the only one. When multi-party private computation is applied in the public sector, sometimes privacy must be compromised in order to accommodate other important social values. The computation can also be applied in the private sector, such as in a business setting. For example, two (or more) companies might want to compute a function cooperatively; however, neither of them wants to share their private information. In both public and private sector applications, it would be beneficial to be able to quantify secrecy so that a tradeoff could be made, for example, between secrecy and computational efficiency. In [5], similar arguments are presented about ideal secrecy and acceptable secrecy. Meanwhile in [2], an information theoretical framework is proposed and two quantitative definitions of secrecy for multi-party private computation are defined, namely, *relative secrecy* and *absolute secrecy*. In this paper, we prove a lower bound for the relative secrecy of protocols that solve scalar product problems. We also propose two refined measurements, *evenness* and *spread*, for quantifying information leakage by multiparty private computation protocols.

The remainder of this paper is organized as follows. We give a short review of related works in Section 2. In Section 3, we present the formal framework proposed in [2]. In Section 4, we present our lower bound proof. In Section 5, we present our extension of the formal framework, and use three examples to explain our new measurements. Finally, in Section 6, we present our conclusions and a short discussion about possible extensions of our model. We also indicate the direction of future work.

2 Related Work

Secure two-party computation was first studied by Yao [11] and extended to the multi-party case by Goldreich et al [7]. Through a sequence of efforts, a satisfactory definitional treatment was found and precise proofs for the security of

multi-party computation were devised . A full description of these developments is given in [6]. The general construction approach is as follows. To securely compute a function, it is first converted into a combinatorial circuit. Next, all the parties run a protocol to compute the result of each gate in the circuit. Both the input and the output of each gate are shared randomly and the final output is also shared randomly among all parties, who then exchange their share of information to compute the final result. Although, this general construction is impressive, it also implies that both the size of the circuit and the number of parties involved dominate the size, i.e., complexity, of the protocol. Note that the size of the circuit is related to the size of the input. Therefore, this general construction is not feasible for real world applications with a large input and/or a large number of parties [9].

The high cost of the general approach for large problems has motivated researchers to look for efficient solutions for specific functions, and many protocols have been developed to solve particular problems. There are specific protocols for general computation primitives, such as, scalar products [1, 10], set union and set intersection cardinality [8], and private permutation [3]. In addition, there are protocols for specific application domains, for example, data mining, computational geometry, and statistical analysis. An excellent survey of secure multi-party computation problems can be found in [4].

Most of the above approaches are based on the notion of ideal secrecy, as observed in [5]. In that paper, the authors ask if it would be possible to lower the security requirement from an ideal level to an acceptable level so that an efficient protocol can be developed. A formal framework based on information theory is presented in [2] in which quantitative metrics of the security level of a protocol are proposed.

3 Framework

As our lower bound proof is based on the formal framework in [2], we include a brief introduction to the framework here. In multi-party private computation, n players cooperate to compute a function. Each player holds some private input that is part of the parameters for computing the function. The goal is to compute the function and maintain the secrecy of each party's private input. Given a protocol, P, we use X_i^P to denote the private input of party i, and msg_i^P to denote the message received by party i. We use information theory to model the amount of information revealed after running P. Note that before running P, none of the parties has any information about the other parties' private inputs. However, after running P, each party may know something about some of the other parties' private inputs because of new information gathered during the execution of P. Let $H_i^P = H(X_i^P)$ denote the entropy of the random variable X_i^P, and $H_{ij}^P = H(X_i^P|msg_j^P)$ denote the entropy of the random variable X_i^P given msg_j^P. The conditional entropy corresponds to the intuitive idea of the amount of information (uncertainty) of X_i^P from party j's perspective after receiving msg_j^P.

Two measurements, *relative secrecy* and *absolute secrecy*, of the secrecy of protocol P are defined as $\min_{i,j}(H_{ij}^P/H_i^P)$ and $\min_{i,j}(H_{ij}^P)$ respectively.

4 Lower Bound

In this section we show that for any two party scalar product protocol, the relative secrecy can not be better than $\frac{1}{2}$. Without loss of generality, let us assume that the protocol proceeds in rounds, where Alice and Bob send messages to each other alternately, with Alice sending the first message. We can record the communication between Alice and Bob as a sequence of messages, $msg = (msg_1^A, msg_2^B, \ldots)$. Given a message sequence msg, we say that an input sequence X of Alice(Bob) is *compatible* with msg if it is a possible record of the communication when the input sequence of Alice(Bob) is X. We use $I_A(msg)(I_B(msg))$ to denote the set of input sequences, that are compatible with msg, for Alice(Bob). Note that msg is a possible record of the communication when Alice's input is in $I_A(msg)$ and Bob's in $I_B(msg)$. We use $I_{A,B}(msg)$ to denote $\{(X,Y)|X \in I_A(msg), Y \in I_B(msg)\}$. The set $I_A(msg)(I_B(msg))$ can be further partitioned into two subsets according to the output value $u(v)$. We use $I_{A,u}(msg)(I_{B,v}(msg))$ to denote the set of input sequences compatible with msg and the final outcome. Note that, for all $X \in I_{A,u}(msg)$ and $Y \in I_{B,v}(msg)$, $XY = u + v$. Here, we consider the case where each number is from $GF(2)$ and the input vector is n dimensional. A general lower bound can be derived by the same approach. Below, we present a high-level sketch of the lower bound proof. If after the execution of the protocol, the information content of the input sequence of Alice(Bob) is still high, it means that many input sequences should be compatible with the recorded conversation. However, a larger $I_A(msg)$ would imply a smaller $I_B(msg)$, since each sequence in $I_B(msg)$ paired with each sequence in $I_A(msg)$ has to satisfy the condition that their scalar product is equal to the sum of their outputs. We therefore derive a lower bound. To formalize the above sketch, we state some basic facts from information theory and linear algebra.

Fact 1
Let X be a random source with n possible outcomes, $H(X) \leq \log n$. In other words, for a random source to have entropy n, we need at least 2^n possible outcomes.

Fact 2 *Let I_1, I_2 be two sets of n-dimensional binary vectors. We use $dim(I_1)$ to denote the dimension of the subspace spanned by I_1.*

- *If $|I_1| \geq 2^k$, then $dim(I_1) \geq k$; and if $dim(I) \leq k$, then $|I| \leq 2^k$.*
- *If I_1 and I_2 are orthogonal, i.e., the scalar product between every vector in I_1 and I_2 is zero, then $dim(I_1) + dim(I_2) \leq n$.*

Given a message sequence msg, let $\mathbf{0_A} = I_{A,0}(msg)$, $\mathbf{0_B} = I_{B,0}(msg)$, $\mathbf{1_A} = I_{A,1}(msg)$, and $\mathbf{1_B} = I_{B,1}(msg)$. By Fact 2, we get $dim(\mathbf{0_A}) + dim(\mathbf{0_B}) \leq n$

and $dim(\mathbf{1_A}) + dim(\mathbf{1_B}) \leq n$. Now consider the relationship between $\mathbf{1_A}$ and $\mathbf{0_B}$. Assume that $dim(\mathbf{1_A}) = k$ and $(i_1, i_2, \ldots, i_k)$ form a basis of the subspace spanned by $\mathbf{1_A}$. Consider the set of vectors constructed by combining an even number of vectors in the basis, denoted by I'. There are exactly $2^{k-1} - 1$ vectors in the set, because the summations of the even terms and odd terms of a binomial sequence are the same. However, the zero vector is not included in our subset. Clearly $dim(I') \geq k - 1$ and the space spanned by $\mathbf{1_A}$ contains both vectors in $\mathbf{1_A}$ and I'. Using Fact 2 again, but this time for I' and $\mathbf{0_B}$, we get $dim(I') + dim(\mathbf{0_B}) \leq n$, which implies $dim(\mathbf{1_A}) + dim(\mathbf{0_B}) \leq n + 1$. If $H(X_A|msg) \geq k_1$, then by Fact 1, $|I_A(msg)| \geq 2^{k_1}$. Without loss of generality, assume that $|\mathbf{1_A}| \geq 2^{k_1-1}$; therefore, $dim(\mathbf{1_A}) \geq k_1 - 1$. Since $|I'| \geq |\mathbf{1_A}| - 1$ and the the number of vectors in the space spanned by $\mathbf{1_A}$ contains every vector in I' and $\mathbf{1_A}$, we derive that there are at least $|I'| + |\mathbf{1_A}| \geq 2^{k_1} - 1$ vectors in this space. Therefore, $dim(\mathbf{1_A}) \geq k_1$. Hence, by $dim(\mathbf{1_A}) + dim(\mathbf{1_B}) \leq n$ and $dim(\mathbf{1_A}) + dim(\mathbf{0_B}) \leq n + 1$, we get $dim(\mathbf{1_B}) \leq n - k_1$ and $dim(\mathbf{0_B}) \leq n - k_1 + 1$. There are at most 2^{n-k_1+1} vectors in the vector space spanned by $\mathbf{0_B}$. However, half the vectors in this space are not in $\mathbf{0_B}$, so we get $|\mathbf{0_B}| \leq 2^{n-k_1}$; therefore, $|I_B(msg)| = |\mathbf{0_B}| + |\mathbf{1_B}|LIA \leq 2^{n-k_1+1}$. If $H(X_B|msg) \geq k_2$, then by Fact 1, $|I_B(msg)| \geq 2^{k_2}$. Now we have $2^{k_2} \leq |I_B(msg)| \leq 2^{n-k_1+1}$. Thus, we get $k_1 + k_2 \leq n + 1$ and the following lemma and theorem.

Lemma 1 *For any two-party scalar product protocol P, if $H(X_A|msg) \geq k_1$ and $H(X_B|msg) \geq k_2$, then $k_1 + k_2 \leq n + 1$.*

Since $H(X_A) = H(X_B) = n$, we get $H(X_A|msg)/H(H_A)+H(X_B|msg)/H(X_B) \leq 1 + 1/n$. The relative secrecy of the protocol is

$$min(\frac{H(X_A|msg)}{H(X_A)}, \frac{H(X_B|msg)}{H(X_B)}) \leq \frac{1}{2} + \frac{1}{n}.$$

Theorem 1 *For any two-party scalar product protocol, the relative secrecy is at most $\frac{1}{2} + \Omega(\frac{1}{n})$.*

5 Extension of the formal framework and examples

Although the two metrics, relative secrecy and absolute secrecy, capture the amount of information revealed by a protocol, they fail to distinguish intuitively apparent differences between various protocols. For example, many two-party scalar product protocols have a relative secrecy of $\frac{1}{2}$, but, it is obvious that a protocol that allows Alice and Bob to send half of their respective inputs to each other is not acceptable. We try to capture the intuition by extending the definition of the secrecy metrics. First we introduce the concept of *evenness* to overcome the drawback of the above-mentioned measurements, which only capture a global view of information leakage. Now consider two protocols, each with relative secrecy $\frac{1}{2}$. In the first protocol, the amount of information leakage only reaches $\frac{1}{2}$ when all the input elements are considered. In the other protocol,

however, the information leakage reaches $\frac{1}{2}$ when only a single input element is considered. Clearly, the first protocol is better than the second. We introduce the concept of *spread* to capture the intuition that the first protocol is better. Before we formally define evenness and spread, we introduce some notations. We present only the two-party case here, and defer the multi-party case to a full paper. Let us first generalize the definition of H_i^P and H_{ij}^P to any subset of input elements. Let A and B denote the two parties. For player A(the definition for party B is similar), let $X_A^P = (x_1, x_2, \ldots, x_n)$, and $S = \{x_{k_1}, x_{k_2}, \ldots, x_{k_r}\} \subseteq \{x_1, x_2, \ldots, x_n\}$. We use $H(S)$ to denote $H(x_{k_1}, x_{k_2}, \ldots, x_{k_r})$ and $H(S|msg)$ to denote $H(x_{k_1}, x_{k_2}, \ldots, x_{k_r}|msg)$. Define $H_A^P(S) = H(S)$ and $H_{AB}^P(S) = H(S|msg_B^P)$. Let $r_A = r = min_S\{\frac{H(S|msg_B^P)}{H(S)}\}$, $rg_A = \frac{H(X_A^P|msg_B^P)}{H(X_A^P)}$, and $\eta_A = rg_A - r_A$. In the above definitions, r_A is the minimum ratio between the information of any subset of the secret input before and after the execution of the protocol, rg_A is the ratio for the whole input. It is reasonable to replace rg_A by r_A; however, we feel it is more informative to define evenness to be η_A, and interpret it as the measurement of the evenness of information leakage about player A. When η_A equals zero, it means that player A's input is leaked evenly. We define the spread for player A as $min\{|S| : \{\frac{H(S|msg_B^P)}{H(S)}\} = r_A\}$; that is, the minimum number of input elements required to reach the maximum information leakage level. An ideal protocol should have relative secrecy as close to one as possible, evenness of every player as close to zero as possible, and spread of every player as large as possible. We use three two-party scalar product protocols to demonstrate the concept of evenness and spread. In the two-party scalar product problem, the two parties, Alice and Bob, have private input X_A and X_B(two n dimensional vectors), respectively. A solution to this problem is a protocol that, after running, enables Alice and Bob to correctly compute the numbers u and v respectively, such that $u + v$ is the inner product of X_A and X_B, i.e., $X_A \cdot X_B$. Let $*$ be the matrix product operator, and X_B^T be the transpose of X_B. Then, $u + v = X_A \cdot X_B = X_A * X_B^T$. Hereafter, we assume that $X_A, X_B \in GF(p)^n$, where $GF(p)$ is a Galois field of order p, and p is a prime number. We also assume that X_A and X_B are uniformly distributed and that both parties are semi-honest, i.e., they both follow the protocol and do not deliberately deviate from it to get more information. Instead, they only deduce information from the messages they receive.

Examples

Our first example is a naive protocol whereby Alice sends the first half of her vector to Bob, and Bob sends the second half of his vector to Alice. It is obvious that relative secrecy $r_g = \frac{1}{2}$, which matches the best protocol. However, it is also obvious that this is not a very appealing solution, because the evenness of this protocol is $\frac{1}{2}$. Thus one party has full information of half the private input elements. In addition, the fact that the spread is equal to one makes the situation even worse.

For the second protocol, we use the Chinese Remainder theorem to encode each element of the input vectors with the same base. Specifically, we pick two consecutive integers, p_1, p_2, such that $p_1 p_2 > p$ and encode each number x as $(x \mod p_1, x \mod p_2)$. Thus, $X_A = ((x_{11}, x_{12}), \ldots, (x_{n1}, x_{n2}))$ and $X_B = ((y_{11}, y_{12}), \ldots, (y_{n1}, y_{n2}))$. Alice then sends the first coordinate of her private input, $(x_{11}, x_{21}, \ldots, x_{n1})$, to Bob and Bob sends the second coordinate of his private input, $(y_{12}, y_{22}, \ldots, y_{n2})$, to Alice. Alice computes $a = \sum_{i=1}^{n} x_{i2} y_{i2} \mod p_2$, and set $u = p_1 p_1^{-1} a$; and Bob computes $b = \sum_{i=1}^{n} x_{i1} y_{i1} \mod p_1$, and set $v = p_1 p_1^{-1} b$, where $p_1 p_1^{-1} = 1 \mod p_2$ and $p_2 p_2^{-1} = 1 \mod p_1$. It is easy to see that the relative secrecy of the protocol is again $\frac{1}{2}$, but this time the evenness is 0, since half of the information of each private input element is revealed to the other party. However, the spread of the protocol is 1; for example, once Bob gets x_{11} the information about x_1 is reduced to about $\frac{1}{2}$.

The third protocol [5] operates as follows. First Alice and Bob agree to an $n * n$ invertible matrix M and a positive integer k that is not larger than n. The rest of the protocol comprises the following steps:

Alice		Bob
1. Compute $X'_A = X_A * M$.		Compute $X'_B = (M^{-1} * X_B^T)^T$.
Let $X'_A = [x_{A_1}, \ldots, x_{A_n}]$,		Let $X'_B = [x_{B_1}, \ldots, x_{B_n}]$,
$\bar{X}_A = [x_{A_1}, \ldots, x_{A_k}]$,		$\bar{X}_B = [x_{B_1}, \ldots, x_{B_k}]$,
$\underline{X}_A = [x_{A_{k+1}}, \ldots, x_{A_n}]$		$\underline{X}_B = [x_{B_{k+1}}, \ldots, x_{B_n}]$
2.	Alice $\xrightarrow{\bar{X}_A}$ Bob	
	Alice $\xleftarrow{\underline{X}_B}$ Bob	
3. $u = \underline{X}_A * \underline{X}_B^T$		$v = \bar{X}_A * \bar{X}_B^T$

In this protocol, M is an n by n invertible matrix. Without loss of generality, let $S = \{x_{A_1}, x_{A_2}, ..., x_{A_r}\}$ and $T = \{x_{A_{r+1}}, ..., x_{A_n}\}$. $H(S) = r * \log p$. Let $msg = \{msg_1, msg_2, ..., msg_n\}$. We have the following linear system of equations from Bob's perspective:

$$\begin{cases} a_{11} * x_{A_1} + a_{12} * x_{A_2} + \cdots + a_{1r} * x_{A_r} + \cdots + a_{1n} * x_{A_n} = msg_1 \\ a_{21} * x_{A_1} + a_{22} * x_{A_2} + \cdots + a_{2r} * x_{A_r} + \cdots + a_{2n} * x_{A_n} = msg_2 \\ \qquad\qquad\qquad\qquad\qquad\qquad \ldots\ldots\ldots\ldots \\ a_{k1} * x_{A_1} + a_{k2} * x_{A_2} + \cdots + a_{kr} * x_{A_r} + \cdots + a_{kn} * x_{A_n} = msg_k \end{cases}$$

$H(S, T|msg) = (n-k) \log p$. Moreover, $H(S, T|msg) = H(S|msg) + H(T|S, msg) = H(S|msg) + \max_S\{(n-r-k), 0\} * \log p$. If $r \leq n-k$, $\frac{H(S|msg)}{H(S)} = \frac{r * \log p}{r * \log p} = 1$. Otherwise, $\frac{H(S|msg)}{H(S)} = \frac{(n-k) * \log p}{r * \log p} = \frac{n-k}{r} < 1$. Therefore, $\min_S\{\frac{H(S|msg)}{H(S)}\} = \frac{n-k}{n}$, where $|S| = r = n$. The relative secrecy for Alice's input is $\frac{n-k}{n}$. The evenness is thus $\frac{n-k}{n} - \frac{n-k}{n} = 0$, and the spread is n. For Bob's input, the relative secrecy is now $\frac{k}{n}$, however, the evenness and spread are the same as for Alice.

6 Conclusion and Future Works

In this paper, by proving a lower bound, we show that revealing half of the private information is unavoidable in two-party protocols that solve the scalar

product problem by only allowing the two parties to communicate with each other. Although this seems intuitively straightforward, proving the claim without the help of an information theoretical formalism is non-trivial. Our lower bound proof not only confirms our intuition, but also demonstrates the advantage of the information theoretical framework. To better capture our intuition, we also propose refinements and extensions of the measurements of information leakage for two-party secure computation. We hope that analyzing protocols formally will not only provide solid certification of the secrecy of existing protocols, but also facilitate the design of better protocols. Using the Chinese Remainder theorem to design protocols is an interesting approach worthy of further investigation. In this paper, we assume that inputs are uniformly distributed. We feel it would be a very interesting and challenging task to develop a method that incorporates players' a priori information about others players' private inputs into the formalism. Finally, and obviously, extending the model to multi-party situations and analyzing some interesting problems is logically the next step.

7 Acknowledgement

This work was supported in part by the National Science Council under the Grants NSC94-2213-E-001-004, NSC-94-2422-H-001-0001, and NSC-94-2752-E-002-005-PAE, and by the Taiwan Information Security Center (TWISC) under the Grants NSC 94-3114-P-011-001, NSC 94-3114-P-001-001-Y, NSC94-3114-P-001-002-Y and NSC94-3114-P-001-003-Y.

References

1. M. J. Atallah and W. Du. Secure multi-party computational geometry. *Lecture Notes in Computer Science*, 2125:165–179, 2000.
2. Yi-Ting Chiang, Da-Wei Wang, Churn-Jung Liau, and Tsan-sheng Hsu. Secrecy of two-party secure computation. In *Proc. 19th Annual IFIP WG 11.3 Working Conference on Data and Applications Security,Lecture Notes in Computer Science, Vol. 3654, Jajodia, Sushil; Wijesekera, Duminda (Eds.)*, pages 114–123, 2005.
3. W. Du and M. J. Atallah. Privacy-preserving cooperative statistical analysis. In *Proceedings of the 17th Annual Computer Security Applications Conference*, pages 102–110, New Orleans, Louisiana, USA, December 2001.
4. W. Du and M. J. Atallah. Secure multi-party computation problems and their applications: A review and open problems. In *New Security Paradigms Workshop*, pages 11–20, Cloudcroft, New Mexico, USA, September 2001.
5. W. Du and Z. Zhan. A practical approach to solve secure multi-party computation problems. In *Proceedings of New Security Paradigms Workshop*, Virginia Beach, Virginia, USA, September 2002.
6. O. Goldreich. *Foundations of Cryptography Volume II Basic Aplications*. Cambridge, 2004.
7. O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game, or: A completeness theorem for protocols with honest majority. In *Proc. 19th ACM Symposium on Theory of Computing*, pages 218–229, 1987.

8. M. Kantarcoglu and C. Clifton. Privacy-preserving distributed mining of association rules on horizontally partitioned data. *IEEE Transactions on Knowledge and Data Engineering*, 16(9):1026–1037, 2004.
9. Dahlia Malkhi, Noam Nisan, Benny Pinkas, and Yaron Sella. Fairplay — a secure two-party computation system. In *Proceedings of the 13th Symposium on Security, Usenix*, pages 287–302, 2004.
10. J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 639–644, July 2002.
11. A. C. Yao. How to generate and exchange secrets. In *Proceedings of the 27rd Annual IEEE Symposium on Foundations of Computer Science*, pages 162–167, November 1986.